ENG5041P
Ioana Susnoschi Luca
GUID: 2146112S

MEng, Biomedical Engineering

Design and development of a tool to facilitate learning to walk with a powered prosthesis

Academic year 2018-2019

Supervisors
Dr. Juan C. Moreno - Instituto Cajal, CSIC
Dr. Aleksandra Vuckovic - University of Glasgow
Dr. Alasdair Clark - University of Glasgow

Special Thanks

The author acknowledges Össur Iceland ehf. for their support with the Power Knee II open development platform donated to CSIC for research purposes

To Andrea Ortiz Cuadros, whose guidance was often needed and always appreciated

To Javier Gil Garcia, for all the time and patience

To Ivan German (Össur Iberia), for valuable insight into amputee retraining and prostheses.

...and to the entire Neural Rehabilitation group at Instituto Cajal, for offering support and brightening up the dullest days.

Abstract

The current paper presents research on a segment of gait analysis and studies to what extent prostheses' designs are able to mimic the behaviour of the lost limb while blending into the locomotion system. This is achieved by producing a tool and associated methodology that facilitates the process of learning to walk with a powered prosthesis. Simulation results of real time event detection flat ground walking at subjects' preferred pace yielded a detection accuracy of 95.0%. Obtaining the time spent on monitored limb as proportion of the gait cycle allows correction of one of the principal sources of gait asymmetry in amputees, with the help of an audio feedback tool. This opens up the potential to construct simple and efficient instrument that targets gait asymmetry. A shift to a more suitable environment for real time gait processing should be considered before enhancing and increasing the complexity of the system.

Contents

List of Figures

List of Tables

1 Introduction

Gait analysis is a huge area of study and engineering solutions to issues ranging from neurodegenerative diseases, such as Parkinson's disease, to traumatic lower limb amputation, are under continuous development. Advancements in technology have a key part on the stage of prosthetic limbs, meaning that research does not only seek to restore basic function of the missing limb; it aims to reintegrate amputees by designing prostheses that actively mimic the behaviour of the lost limb and blend into the locomotion system.

The aim of the Dissertation has been to produce a tool and a methodology associated with it, to facilitate the process of learning to walk with a powered prosthesis, for which the motorised robotic knee Power Knee II had been chosen, due to the possibility to access data from its embedded sensors. Although robotic transfemoral prostheses provide power, resistance and locomotion mode recognition, some of the fundamental problems in adapting to gait with an artificial limb persist. Therefore, the objectives of this research are centred around the process of gait retraining following prosthesis fitting, which has been investigated in many publications, but is yet to be implemented. Since rehabilitation techniques still rely heavily on visual assessment of gait, which introduces variation stemming from professionals' subjectivity, a system that is easy to set up and targets pervasive defects could provide additional help that amputees can use outside the clinical environment. Moreover, this objective evaluation of basic walking patterns may contribute to a unified protocol for rehabilitation that allows measuring and grading the gait deviations in amputees fitted with a new prosthesis.

The work presented in this Dissertation is the result of a study that started with assessing the evolution of the design, prototype and testing of lower limb prostheses, based on research published between 1974 and 2018. Identification of one ubiquitous issue that could be prevented or corrected in amputee gait followed review of literature and data analysis from an amputee recently fitted with Power Knee. The novel methodology would access load cell, knee angle and knee angular velocity sensors from the robotic knee to determine whether the amputee spent less time on the prosthetic limb compared to the healthy one - a principal source of gait asymmetry in amputees, and would provide audio feedback accordingly. However, after having studied the process and outcomes of gait retraining with a variety of passive, semi active and robotic prostheses, it was decided that the instrument could be designed to comprise wearable sensors such as the inertial units and force sensing resistors, and offer a solution that can be applied to a wide range of prostheses and ortheses. Existing literature covering the use of biofeedback to help the integration of the prosthetic with the amputee tackles the challenges faced by the latter in carrying out activities of daily living and mainly presents assistive solutions. By identifying one of the root problems in gait symmetry - stance phase duration on the healthy leg, it is possible to develop a tool that helps prevent or reduce its impact on the amputees' gait. The first phase after the construction of such a methodology is its testing as part of a feasibility study involving healthy individuals, to offer an insight into potential issues with implementing the system as part of simple walking tasks.

2 Literature Review

2.1 Human gait

Gait is a cyclic movement that relies on leg support and movement of lower limb segments to displace the body from one position to the following, the result of intricate communication pathways between the central nervous system, joints and muscles [2], involving sensorimotor pathways in order to obtain balance, posture and stability in motion. The broadest classification of gait recognises two classes: ballistic (air-borne) gaits, characterised by a phase during which neither foot touches the ground and by a bounce-like trajectory of the body's centre of motion, and walking gaits, during which at least one leg is in contact with the ground [4] and the centre of gravity of the body has a flatter trajectory.

Five principal motions occur over the course of one gait cycle - rotation of the pelvis, pelvic tilt, knee flexion, ankle flexion and toe flexion [5]; additionally, secondary movements of the arms and trunk posture are involved in order to maintain balance and decrease energy expenditure [6], which become increasingly important in fast walking and running. Bipedal gait is made up of two main phases, or periods - stance phase and swing phase, in respective proportion of 62-38% on each leg. During the overlap of the two periods, the body is supported by a single limb (single leg stance), while the remaining 25% is double stance phase. The periods are further divided into sub-phases delimited by gait events and are briefly described below and shown in Figure 1.

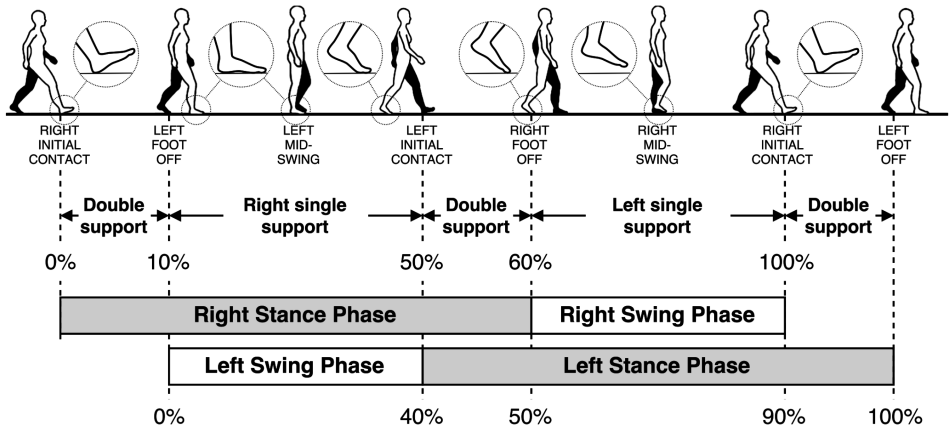


Figure 1: Gait events and phases of gait, including the percentage of one cycle [1].

2.1.1 Stance phase

During stance phase, the longest of the two main phases, at least one foot is on the ground, supporting between [this-that] of body weight. Stance phase comprises four gait events:

Initial Contact (IC)
This is the first contact between the foot and the ground and occurs between 0-2% of the gait cycle duration, when the heel strikes the ground (hence also named Heel strike); kinetically, the force vector acts almost perpendicularly to straighten the hip and knee [7]. The ankle joint has neutral position and the knee is flexed by approximately 5°[8].

Loading response
Immediately following IC, the load vector points backwards, the foot is lowered to the ground with the hip maintaining the same angle, the knee flexed through 20°and the ankle performing 10deg plantar flexion, to provide shock absorption and decelerate the centre of gravity. Loading response takes the next 10% of the gait cycle [7, 8]. As these occur, the opposite leg enters swing phase, therefore single limb support begins.

Mid-stance (MST) During mid-stance, happening between 12-30% of the gait cycle, the foot is flat on the ground, the hip is in neutral position, the knee extends from 20°to 5°flexion and the ankle joint flexes through to 5deg dorsal flexion, while the centre of gravity moves towards the forefoot.

Terminal stance
From 30% to 50% of the stride, the heel lifts off the ground moving the ankle to 10deg dorsal flexion; the hip is hyperextended, pushing the centre of gravity of the body forward. Just before half of the stride time has passed, the opposite heel makes contact with the ground.

Pre-swing
The body is supported by both limbs from 50% to 62% of stride time. The knee is flexed passively to 40°, the ankle reaches 15deg plantar flexion exerting power, and force through the font of the foot increases acceleration, as load is shifted onto the opposite leg. This phase ends with the gait event named toe off (TO).

2.1.2 Swing phase

The trajectory of the limb during swing is mostly determined by gravitational components and inertia, while the muscle activity is diminished [9, 10, 11], lowering the energy required for locomotion. Limb swing is considered to be driven by passive dynamics.

Initial swing
From 62% to 75%, the knee flexes to a maximum of 70°, the ankle returns to only 5°plantar flexion and the hip reaches 15°flexion. Although all three joints act to obtain ground clearance, knee flexion of at least 55°is the most important factor. During this phase, muscular input is lowered.

Mid-swing (MSW)
Mid-swing begins when the centre of the two shanks align, at 75% of the gait cycle and terminates when the tibia is in vertical position, in front of the opposite leg. The ankle is back to neutral position and the hip is flexed to 25°; the knee starts extending. In MSW, the lower limb muscles have very little activity.

Terminal swing
At 87% of gait cycle, the knee is extended to neutral position, slowing down the leg, and prepares for the following initial contact.

2.1.3 Balance during gait

Movement has been described by Zambabieri in [29] as a chain of successive postures accomplished by the body segments, under control of the central nervous system. Balance, which comprises body posture dynamics that prevent falling, stems from body segment inertia; thus it is a requirement for successfully carrying out movement, and it is achieved when the centre of mass of the body falls inside its support surface [17]. During flat ground walking, gait cycle comprises of 60% stance phase, most of which is single leg support, which is why regaining balance after a lower limb segment amputation is key to performing ambulation and Activities of Daily Living.

The three systems that ensure balance during movement and walking are the visual system, which aids in movement planning and obstacle avoidance, the somatosensory system, in charge of velocity and position sensing and gravity orientation, and the vestibular system, which senses acceleration. The somatosensory system is comprised of afferent neurones, with nerve endings located in the leg. Amputation of the lower limb and lack of proprioception translate into missing information about the sense of force developed, joint motion and position in space [20]; this modifies the interaction between these systems and drives the need for a reorganisation of the motor control strategies. A mediator is then required to connect the external environment to the leg stump, which causes desynchronisation between the legs and asymmetric gait [18].

Amongst the gait defects caused by lower limb amputation, the most common ones are asymmetrical weight distribution, forward shifting of the body [20], extended time in stance phase and decreased swing time for the intact limb [23], uneven step length taken with prosthetic limb, decreased cadence and velocity [21]. Moreover, Buckley et. al. have documented differences in amputees' adaptive gait when dealing with obstacles, including reduced approach velocity, decreased foot placement distance on either side of the obstacle, reduced foot clearance and knee flexion of the leading limb [66], while Kavounoudias et. al have observed sensory changes in the nonamputated leg, indicating post-amputation adaptations of the central sensory system [65]. Studies on standing balance indicate significant deterioration of balance in people with lower limb loss[19], but confirm that its partial restoration is possible with rehabilitation and training approaches that involve sensory feedback [22].

2.2 Healthy gait biomechanics and mechanisms

Bipedal gait involves three main joints for displacement, namely the ankle, knee and hip joints, whose coordination and mobility is essential for locomotion. Other parts of the human locomotor (musculoskeletal) system, such as the thorax, upper limbs and feet, also play important roles; however, these are beyond the scope of the present work.

The cardinal axes and planes of movement will be referred to using the names presented in the sketch in Figure 2.

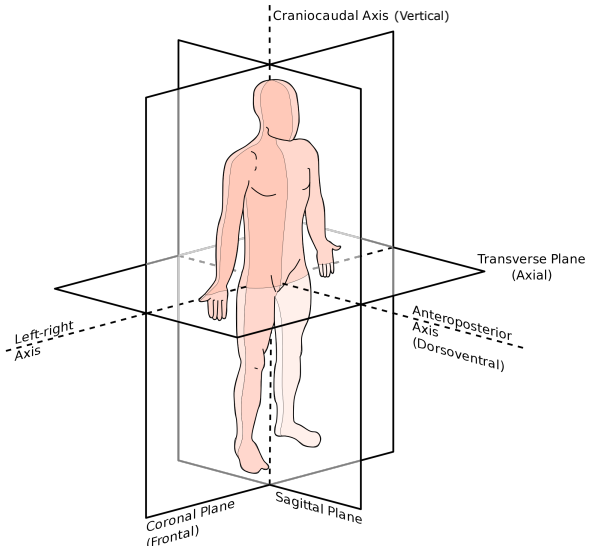


Figure 2: Anatomical planes and axes. Sketch from [64].

2.2.1 Ankle joint biomechanics

Throughout the gait cycle, the ankle joint performs dorsiflexion and plantarflexion alternatively, and acts to move the body centre of gravity forward by producing power at heel off, to absorb shock from initial contact to foot flat and to facilitate forward leg swing[12, 39] by allowing foot clearance. In stance phase, the ankle experiences the ground reaction force from supporting the body load, while in swing phase, inertial force acts on it.

2.2.2 Knee joint biomechanics

The knee joint is formed by the biggest bones of the lower limb- femur and tibia, and has a major role in stability during stance phase and in displacing the leg forward during swing phase [13]. For this reason, its role in healthy gait is paramount. The human organism is characterised by its efficiency in energy usage, which is the result of hundreds of thousand of years of evolution. From the kinematic point of view, efficiency is achieved by limiting the vertical displacement of the centre of mass of the body. As such, it is natural that the whole body is involved in keeping human locomotion as economical as possible. The collection of adaptations taken for this purpose was first described by Saunders in 1953 [14], referred to as *Determinants of gait*, detailed below.

It was thought that, from the kinematic point of view, energy expenditure during walking was kept to a minimum by limiting the vertical displacement of the centre of gravity (CoG) of the body, and that the whole body/lower body was involved through the "Six gait determinants" described by Saunders et al. in 1953 [14]. These are achieved by 1) 4°pelvis rotation about the direction of walking to increase step length, 2) pelvic tilt on the side in swing phase to restrict vertical CoG displacement, 3) 15°knee flexion in stance phase, which describes the CoG trajectory as the arc of a circle, 4) knee extension and foot dorsiflexion at IC to increase limb length, 5) foot plantar flexion at TO for levelling the CoG trajectory. The sketch in Figure 3 is based on the theory of knee-ankle-foot mechanism (3, 4 and 5) that acts to level and smooth the COG progression by increasing the limb length both at IC and at TO. Lastly, the role of the angle between the tibia and femur that places the feet below CoG was though to reduce lateral hip excursion, thus

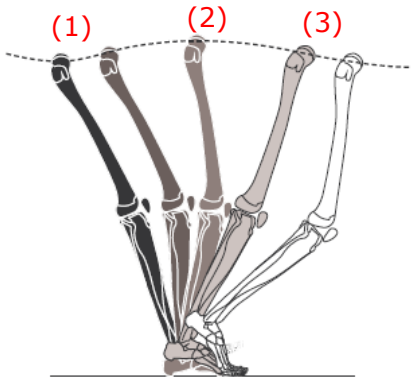


Figure 3: The theoretical effect of Saunders' gait determinants in lengthening the limb and smoothing out the body's CoG trajectory and wight shifting. At IC, the knee is extended and foot is dorsiflexed (1), to make contact with the ground more gentle; the knee flexes (2), maintaining a flatter arc during MST; knee extension and foot plantar flexion (3) . Figure adapted from [13].

energy. However, the first three 'determinants' theories were mostly dismantled through experimental studies that demonstrated little to no effect of pelvic tilt and knee flexion in stance phase on vertical excursion of body COG, and that pelvis rotation did not smooth out the COG trajectory [32]. Moreover, voluntary reduction of the vertical displacement of CoG is energetically costly as it requires increased knee flexion in stance phase, which reduces opposite leg ground clearance, introducing the need for increased angular displacement of the knee and ankle.

Inverted pendulum model provides a more accurate description of kinematic parameters, based on the exchange between kinetic energy and gravitational potential energy. Here, the almost straight stance leg provides the pillar for the arc-like swing leg motion [15]. An example of this is the double stance phase, when the COG is at its lowest point, therefore the potential gravitational energy is the lowest, while the body reaches maximum velocity, equivalent to maximum potential energy [13]; the shift is made at the beginning of swing phase, when the body is raised and velocity decreases. Moreover, an exchange of energy between proximal and distal segments of the body and of the lower limb was proven in [68]; this occurs through muscles, which store and release elastic energy, and articulations. Other means of preserving energy during gait are modulation of frequency of steps [5], which is related to walking velocity and stride length, and the roll-over shape (ROS) performed by the foot during the stance phase, with effects on the balance, kinetics and kinematics of gait [16].

The muscles involved in each gait phase are highlighted in Figure 4. Forward progression of the body during gait is mostly covered by ankle plantar flexors, while support relies on the plantar flexors in single stance, to which knee and hip extensors are added in double stance phase [57]. These are supported by findings from Lim et al. [56], who have proven that the principal five muscles performing the above mentioned are the gluteus maximus, gluteus medius (hip extensors), the three vasti (knee extension), gastrocnemius and soleus (plantar flexion). A transtibial amputation would affect the function of the last two, while in the case of transfemoral amputations, the vastus muscle group is also partly removed, which would entail that forward progression and vertical support relies almost entirely on the gluteus muscles. Seroussi et al. [59] have found that the intact limb of TFA produces more power and the joints are subject to higher torque in order

to aid in gait progression and keep upright stance, as the muscle support requirement is not met.

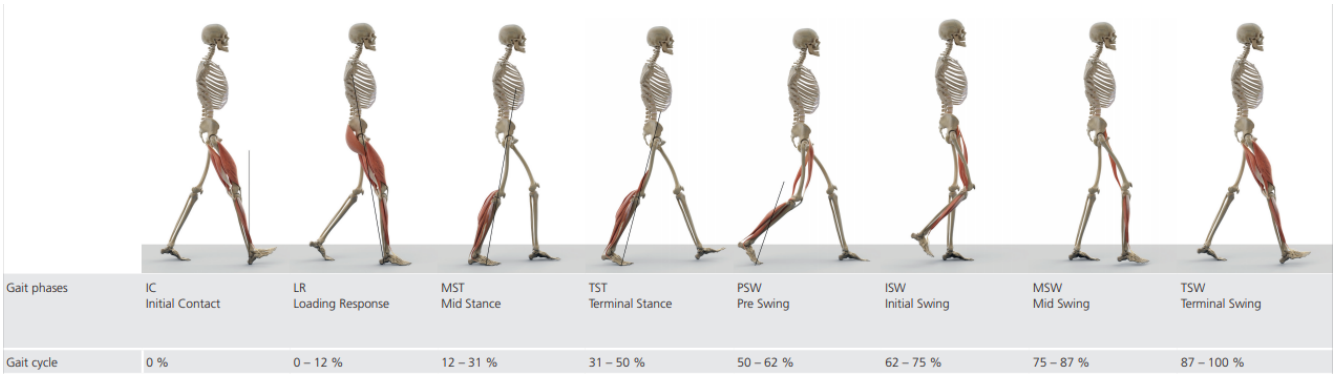


Figure 4: Gait phases shown in relation to the leg muscles involved and ground reaction force vector [8].

Missing limb and implicit proprioception loss may interfere with at least one of the above mentioned mechanisms, depending on the level of amputation, thus affecting the body's capabilities to move in an energetically efficient and almost effortless way. Therefore, gait rehabilitation plays a central role in restoring as much as possible of the natural gait motion. Rehabilitation strategies comprise hip motion level, prosthetic limb motion control, and whole body thrusting, all of which target various walking defects.

2.3 Gait pathology and energy expenditure in lower limb amputees

Below knee amputees

Longer residual limb and full thigh muscles can still achieve a fair level of control, but the missing ankle joint and foot result in improper or incomplete ROS (roll-over shape) in foot progression. The residual ankle dorsiflexors, located on the frontal aspect of the leg, cannot achieve complete foot clearance, which leads to hip hiking in swing phase [28]. A study carried out on passive TTA (trans tibial amputee) prostheses [67] has shown that the walking velocity of below knee amputees is approximately 30% lower than that of healthy individuals, which leads to an increase in stride length and in healthy limb stance phase. Moreover, asymmetry is more evident in subjects with shorter residual limbs.

Above knee amputee (TFA)

Transfemoral amputees (TFA) lack the knee joint, which is the largest and most complex joint in the body. With increasing severity of the amputation (shorter stump length), more hip power is required to compensate for the lack of control, therefore the hip motion pattern is affected, leading to trunk motion changes. In non-automatic prostheses, lack of knee flexion at MST causes vaulting on the contralateral side, which increases the time spent on the healthy limb, altering weight shifting. On the other hand, knee extension in most automatic prostheses, such as Össur's Power Knee, has to be triggered by hip motion control [**IvanOssur**]. For this reason, hip motion level rehabilitation is necessary in order to gain sufficient muscle strength and learn how to trigger the different prosthesis modes, especially in high level TFA, where almost 100% of prosthesis control

comes from this.

Aside from gait compensatory mechanisms, transfemoral amputees were shown to exhibit abnormal proprioception in the residual thigh muscles (quadriceps and hamstrings), which leads to alternative sources of sensing the environment - hip joint position, pressure at stump/socket interface and position of prosthesis [58].

2.4 Use of lower limb prostheses

Although the use of conventional passive prosthetics is heavily documented compared to microprocessor controlled or robotic prosthetics, which, due to the systems' complexity, need additional training time, there are good reasons for a shift to be made towards the latter ones. Passive prostheses do not provide power or store energy, thus push-off relies solely on the user, who has to exercise voluntary control that increases the cognitive load, and exert between three and four times more hip torque during stance, compared to healthy gait [30]. Energetically-passive prostheses increase metabolic power consumption and cause an asymmetrical gait pattern [31], which has been shown to improve when a microprocessor controlled knee replaces the mechanical one [33]. Robotic knee and ankle prostheses have much higher power requirements compared to the leg muscles; nonetheless, technological advancements in Li-ion batteries, magnet brushless motors, semiconductors and sensors based on microelectromechanical system technology have enabled their development and use [30]. Due to these, the biomechanical response of the prosthetic leg is closer to the healthy human leg, supplying energy absorption and return, propelling the body forward and absorbing shock, to some extent. A complete robotic leg prosthesis with three levels of control is comprised of an electric motor, sensors that relay information on joint angle and angular velocity, inertial measurement units and may contain force and torque sensors [30]. Most sensors embedded in robotic prostheses have the purpose of sending information from 1) cortical or neuromuscular activity, 2) posture, 3) locomotive state, or 4) physical interaction between environment-device, to the control system for walking or intent prediction [26].

Lack of confidence in using robotic leg prostheses stems from insufficient training, complexity of control and leads to the above mentioned compensatory mechanisms, which form abnormal gait and cause secondary health issues (e.g. back pain, knee osteoarthritis) [26]. Short term adaptation of gait, such as the one that occurs when transitioning between terrain types, is well documented in the context of lower limb prostesis use. Nonetheless, it is the long-term adaptation, based on sensorimotor learning, that diminishes the maladaptive gait patterns [27], with the help of early fitting of the prosthesis and rehabilitative treatment [22]. Despite the huge developments in robotic prosthesis technology, it is recognised that active prostheses are far from restoring locomotion capabilities to their fullest, and that control strategies employed in current devices do not offer the smooth use that is characteristic of human gait [47]. Therefore, the rehabilitation and learning processes are of paramount importance to restitute the balance and confidence to amputees. Integration of biofeedback tools into the existing rehabilitation techniques has proven successful for gait and balance improvement [22, 23].

The ground pressure sensation transmitted from the foot has a major role in limb coordination for walking, running and performing other activities, and it is one of the features that prostheses fail to restore to amputees. For this reason, non-invasive biofeedback tools have been developed

to deliver sensory feedback and close the physiological control loop that allows safe and efficient gait [22]. The three lines by which this can be delivered are the visual, the auditory and the tactile channels [26], with research evidence showing that simultaneous stimulus delivery to two channels is possible. For the design of a feedback system that is effective in aiding the amputees' rehabilitation and in restoring their locomotion capabilities, it is essential to find the amount of information that, when delivered to the prosthetic device, will improve the interaction without increasing the cognitive load [26]. Motor control and researchers' experience in gait analysis play an important role in selecting key information that will be delivered to the user, and the means by which this will be done [26].

2.4.1 Biofeedback for amputee gait training

Acoustic & Audio Feedback

Acoustic and auditory feedback can be delivered under three forms: an alarm that signals the exceeding of a threshold, a movement error cue about deviation from the target parameter, or a sonic representation of movement variables, where the variable change over time is translated into audio modulations [62]. The auditory signals delivered may use variations in volume, pitch and timbre to indicate variable change [62]; however, these modulations have to be carefully designed to prevent overloading of the auditory path. Auditory feedback in the form of movement error has been employed to signal interior or exterior foot deviation [22], as auditory alarm to indicate success or failure in directional balancing of the body [63], and as representation of movement variables, together with visual feedback, to indicate the amount of load placed on prosthetic the foot [23]. Results of all methodologies reviewed reported improvements in gait: increased gait symmetry and cadence and better dynamic postural control from the audio-visual feedback [23], balance improvements as a result of the auditory alarm [22]. Evidence of auditory cues implemented as early as 1982 [25] shows its usefulness/effectiveness in relaying information on period of swing and knee extension in transfemoral amputees using passive prostheses.

Visual Feedback

Visual feedback is by far the most documented modality of providing sensory information to amputees, and not only. Zambabieri described a visual biofeedback tool to help amputees maintain their balance by projecting the trajectory of their centre of pressure on a screen in changing colour if the CP fell outside of the prescribed area [29]. This simple solution proved effective in providing the amputee with details on their movement. However, it was noted that this method required the user to be more aware in order to be able to correct their position while looking at the screen. Although there have been reports of patients' preference for visual feedback over auditory and haptic one [26], the visual channel is much more susceptible to overloading due to the fact that vision already plays a central role in locomotion. Concurrent feedback displayed in acoustic or vibrotactile form has been suggested as alternative to prevent this.

Tactile or Haptic feedback

Tactile feedback can be applied with frequency, strength, pattern and duration variations [26]. The two methods by which it is accomplished are magnitude-based feedback (stimulus intensity increases as ground reaction force increases) and event-based feedback (stimulus is activated when gait events such as toe off and initial contact are detected) [60]. When using the prototype of a haptic feedback tool that employs gait event detection and lateral skin-stretch, a healthy subject has recognised the location of stimuli on the thigh with 98% accuracy. Vibrotactile stimulation

with pattern changes informing the user on ankle joint position has been tested with two transtibial amputees, showing promising results [61].

In the research encountered, patients' preference swings between visual and auditory feedback [23, 26]. Although results of tested vibrotactile feedback systems indicate that some subjects are successful in decoding haptic stimuli, these systems require more accommodation time and the development of more complicated stimulus systems. Visual feedback systems do not always come in portable form [26], introducing the need for sophisticated equipment and increasing the cost of training and rehabilitation. Moreover, overloading of the auditory path is less probable and has milder consequences than that of the visual path. Clarity of the instructions is one of the main requirements for the development of an effective biofeedback tool. Auditory feedback can be an inexpensive and easy to learn modality shown to enhance patients' confidence in ambulation with a prosthesis and to reduce maladaptive gait patterns. [23, 26, 25].

2.5 Gait analysis techniques and instrumentation

Healthy human gait consists of repeating motion patterns and is considered to be symmetrical - despite research studies proving functional asymmetries due to side dominance in normal gait [3] - with a phase difference of 180°between legs [16], therefore it is possible to approximate human walk with a single stride, which is defined as the portion of gait delimited by two consecutive occurrences of the same gait event. As a convention, the gait cycle begins with the initial foot contact, or heel strike.

Analysis of human gait involves inspecting movement patterns associated with walking and it has long been used to characterise motion of elite athletes. With advanced technological solutions being widely available nowadays, the area has evolved immensely, such that instrumented three dimensional parameter evaluation is possible. Gait analysis is now common practice in clinical settings for rehabilitation, to describe pathology, to assess functional limitations and training outcomes, as well as in research laboratories [53]. However, observational-based assessment is still the core of clinical gait analysis due to the high cost associated with tools required for this, and their maintenance, in spite of existing proof that quantitative gait analysis can improve treatment protocols and outcomes.

Ethier and Simmons consider complete gait analysis to be comprised of four types of physiological data, namely electromyography (muscle electrical signals), anthropometry (dimensions of human body), kinetics (forces that produce motion) and kinematics (motion study, regardless of forces) [5], although the research area is most commonly limited to measurement of kinetics and kinematics of joints [53].

2.5.1 Kinetics

Gait kinetics evaluate the forces that [arise] at the contact between the lower limb and ground during motion. Kinetic analysis is mainly performed using force platforms, which can be complemented with/by instrumented foot platforms and pedometers [13]. A force plate contains one or more piezoelectric sensors or load cells distributed across a flat surface [49] placed between two rigid plates the size of large floor tiles. Changes in posture related to movement are determined by dynamic variation in the ground reaction force [51], which is translated into electrical signals

using electrical transducers, thus obtaining the quantitative three dimensional force distribution (medio-latral, anteroposterior and vertical directions), as well as the three components of torque. Due to their accuracy and completeness, measurements derived from force plates constitute the gold standard in kinetic parameter assessment [50], specifically for balance, often used in research laboratories for rehabilitation, neurology and sport performance [48]. Nonetheless, force plates are expensive and non-portable, and require designated space.

Instrumented foot platforms or insoles, a popular alternative, use pressure transducers that can offer high resolution information on the force between foot-shoe, depending on the size and number of sensors used [13]. Their placement on specific locations on the foot allows determining the vertical foot loading pattern during gait. The Force Sensing Resistors vary their resistance with load applied: when force is exerted on the sensor, the piezoresistive substance inside is compressed and forms a conduction path, increasing the current that passes through. The voltage difference resulted from this is indicative of the amount of applied load [52]. In recent years, robotic prostheses have been equipped with embedded sensors that monitor kinetic parameters, such as the load cell, in order to have an accurate measure of the vertical force applied and thus determine the phase and mode of walking.

2.5.2 Kinematics

Kinematics capture successive motions of the human body, enabling study of complete gait cycles, in particular by use of optical motion capture systems (OMCS), which are considered the gold standard in this area [50]. Apart from gait analysis for diagnosis and rehabilitation purposes, kinematic parameters have lately been used for neural prostheses and functional electrical stimulation control [37]. Video motion analysis systems entail similar issues to force plates: despite their accuracy and the big amount of data provided, their use is limited to laboratory settings as they are expensive and non-portable; in addition, OMCS require calibration before use [53]. This has led to an increase in popularity of alternative kinematic parameter capture instruments, such as accelerometers, magnetometers and gyroscopes, often combined into inertial measurement units (IMUs), goniometers or electrogoniometers and inclinometers [54], all of which are wearable and offer estimation of position and orientation.

Gyroscopes

Gyroscopes are devices that measure angular velocity using the Coriolis acceleration of a vibrating device, and offer the output in radians/second or degrees/second. Angular velocity measurements using only gyroscopes and subsequent computation of secondary quantities (joint angles) offer reasonable accuracy in detection of locomotion parameters. Miniature gyroscopes for gait event detection have shown promising results, with Aminian et.al. employing them for long period detection of toe-off and heel strike from shank angular velocity, as a monitoring tool, and Coley et.al. developing an algorithm that uses a single miniature gyroscope to detect stair climbing [45]. A study investigating the suitability of gyroscope sensors for control of Functional Electrical Stimulation has shown that gyroscopes capture both stance and swing phase information, although foot switches are needed for recalibration of the former, to prevent signal drift [46]. These systems generally have the advantage of low power consumption, therefore an extended battery lifetime, which makes them suitable for out-of-lab monitoring applications.

Accelerometers

Accelerometers measure inertia of a body and have recently gained popularity in body worn monitors, due to the increasing need for out-of-laboratory measurements, including age-related conditions, monitoring and assessment. Low cost body-worn accelerometers are used for applications such as monitoring of postural control and sway, with proof from studies by Mancini et.al. in [36] that body-worn accelerometers can be employed for this purpose as a lower cost, efficient substitute to force plates. An accelerometer attached to a gait belt was used by Rine et. al. to measure balance in relation to vestibular function [34]. A similar instrument and set up were used for Parkinson's Disease symptom assessment [38].

Raw accelerometer and gyroscope signals can be combined for gait event detection, to quantify daily activities [37] and identify pathology or deviations from normal walking patterns. Additionally, secondary quantities such as velocity, displacement and joint angles can be obtained by performing integration of raw signals, while position and orientation information is available from three-dimensional sensors (6 degrees of freedom IMUs) [55].

Inertial Measurement Units

IMUs are usually comprised of a three axis gyroscope, a three axis accelerometer and, optionally, a three axis magnetometer. In health-related applications, the use of IMUs includes biomechanical studies, rehabilitation devices, movement and exercise analysis [40], but is also implemented for providing feedback to sensory-impaired people. They can be employed as single or multiple units, depending on the depth of data that is needed. Nonetheless, assessment of lower limb rehabilitation exercises in [42] did not differ in accuracy between using one, two or three IMU sensors - the system was successful in discriminating between correct and incorrect performance. A study by Leardini et.al. employed multiple IMUs for visual biofeedback to aid in orthopaedic rehabilitation and obtained suitable accuracy and sensitivity of the system detection of angular measurements [41]. Inclusion of a 3D magnetometer results in a drift-free estimation of the body's orientation, as indicated by results from [43]. On the other hand, drift reduction using joint limit constraints is presented in [44], for leg posture estimation using an extended Kalman filter.

3 Eurobench - Gait with lower limb prostheses

A study was performed based on evaluation of 250 papers published between 1974-2018, that have been found by a systematic literature search on Scopus database, to describe the state of the art in benchmarking bipedal locomotion skills of lower limb robotic prostheses. The keywords and wildcards used for the database search can be found in Appendix, 10.3. The relevant publications were analysed by reading the Abstract, Materials and Methods, Results and Conclusions, and were evaluated with respect to the specific motion tasks addressed, as well as the different performance criteria considered for the motions. The motion tasks were assigned to one of the three sub-categories: 1) different types of walking motions (e.g. flat ground walking, on treadmill, on slopes); 2) Balancing tasks while standing (e.g. single leg); 3) other tasks, such as kneeling, sit-to-stand movement. The distribution of key indicators of prosthesis performance, the kinetic and kinematic parameter evaluation and the interaction between human and prostheses among the papers surveyed was quantified.

It was found that the top four motion tasks evaluated experimentally in literature are flat ground walking (54.4% papers), treadmill walking (26.4%), walking on slope (19.2%) and stair negotiaion (18.8%). Assessment of balance during motion is sparse, found in only 2.51% papers, while running an sit-to-stand transitions were tested in 6.69% papers. The top stated goals of the prostheses covered in research were Ability to minimise failures (23.2% papers) and Energy efficiency (22.4%). Kinematic and kinetic parameters of the prosthesis are evaluated in 58.8% and 50.8% publications respectively, but only 2.0% evaluate total body coordination. It became apparent that the trend in recent years has shifted towards development of new technologies, rather than validation of existing ones. Only a small number of the publications describing prototype or initial phase designs are followed by more detailed studies, or reevaluate the products on a greater sample size. Rather, most research published in the past 10 years presents prototypes or preliminary models for prosthetic limbs and only assess the two most documented motion tasks: flat ground walking and treadmill walking.

While the majority of publications state that their primary aim is to replicate healthy human gait as closely as possible, only 7.6% of them analyse this specific factor. Although the essential performance indicators, such as kinetic and kinematic parameters or metabolic expenditure need to be objectively quantified, the development of a prosthetic limb should be centred on its interaction with the user. This can be achieved by introducing questionnaires or focus groups in the research protocols and has the potential to expose major faults in robotic limbs that otherwise show encouraging results.

Despite similar performance indicators being assessed in most of the 250 publications, none of them mentions using or creating a database for the studies. This leaves major room for improvement, especially in the case of studies in which normal gait data is collected from healthy subjects, following very similar protocols. The scientific community would benefit from a) standardized protocols and performance indicators for data collection and b) a database containing the information acquired using these protocols. This would not only decrease the resources needed to conduct a study, but it would also reduce inter-study variability, leading to more accurate and reliable research.

4 Methods

The present study is a feasibility study that aims to investigate gait patterns in healthy subjects and amputees, so as to aid in the development of a methodology comprising a biofeedback tool that facilitates gait rehabilitation in robotic prosthesis users, and that can be [extended] to passive orthotic devices.

4.1 Hypothesis

Lower limb amputees who learn how to use prostheses have difficulties in developing a correct walking pattern without [proper] rehabilitation, for which the right instruments and professionals are required. Irrespective of the type of prosthesis (robotic, semi-active or passive), follow-up studies have revealed compensatory mechanisms such as diversion from the 60-40% gait phase proportion, increased hip circumduction and/or absence of knee extension in stance phase, all of which lead to secondary conditions. The introduction of a biofeedback system that closes the information transmission loop between the central nervous system (CNS) and the missing extremity can have a positive impact on the rehabilitation outcome by targeting gait pathology types individually, and reducing the involvement of the personnel/ reducing reliance on visual analysis of gait.

The development of the audio feedback tool for gait training is based on the hypothesis that informing the user of the correct duration of the stance phase on the prosthesis can decrease one of the fundamental and most encountered problems when walking with an artificial leg: timing of weight shifting. Following detection of the gait phases from the angular velocity and sagittal acceleration of the shank, the developed software computes the threshold value for stance phase and informs the patient when they should shift their weight on the opposite limb.
The objectives of the present study are:

- to revise the state of the art on methodologies to validate gait with active prostheses
- to determine to what extent gait with a powered and semi-active prosthesis differs from normal gait, in experimented users
- to develop a measurement system for recording powered prosthetic data during gait conditions
- to design, test and validate a biofeedback tool for gait training for lower limb prosthesis users

4.2 Gait with a robotic knee prosthesis

More insight into transfemoral amputee locomotion with active and semi active prostheses was gained through personal communication with the Technical Support and Sales Representative of Össur Iberia and by studying gait patterns from one robotic TFA prosthesis user (Power Knee II, Össur) when walking on flat ground, at their most comfortable speed. At the time of data recording, the prosthesis user's age was 38 years, their height was 173cm and their weight was 53kg. The available data was limited due to the lack of users of robotic prostheses in the area of the receiving institution.

The Össur Iberia representative, a transfemoral amputee themselves, was contacted in November and the aims and objectives of this study were presented to them in order to confirm the above stated Hypothesis. Given their involvement in active, semi-active and passive lower limb prostheses fitting, they were able to provide more information on how the amputee population would benefit from an audio feedback tool. Furthermore, they offered information on the process of gait training for new prosthesis recipients, including the variety of rehabilitation techniques that are offered at present, as well as their weaknesses from the point of view of rehabilitation outcome.

Power Knee II is a robotic knee joint that employs the **6** embedded sensors that sample the state of the prosthesis at $F_S = 100Hz$ to identify the current locomotion mode and actively flex or extend the knee during flat ground walking, provide assistance for sit-to-stand transitions and increase the resistance in stair descending [70]. Wireless connection to the prosthesis is established using the dongle in Figure 5. PowerLogicII, a PC application provided by Össur, allows accessing the sensor data and setting the following knee parameters: 1) experience level; 2) lower limb anthropometric data; 3) intensity of warning vibrations. Data logging for post processing is performed using BxTester, an application designed for testing of the prosthesis. Power Knee II is equipped with two 3-axis accelerometers (shank and thigh), two single axis gyroscope (shank and thigh), a load cell located at the bottom of the prosthesis, and a knee torque sensor. Secondary parameters are derived from these, such as knee position and deflection angles, shank and thigh velocity.

The variables that were selected for analysis were load cell data, knee joint angular velocity in sagittal plane and knee angle, as they give a complete representation of the weight shifting process in a gait cycle. Knee flexion reaches a maximum of 60-70°at initial swing and starts decreasing to prepare for heel strike, when it is in full flexion, therefore IC was found as the first zero on the knee angle plot, following the maximum value. The robotic prosthesis uses the load cell to identify the terminal stance and provide the power necessary for leg swing, thus the velocity of the knee joint is the highest at TO event. The analysis results are presented in Section 5.1.

4.3 Materials

Data was acquired and processed on an Asus Pro P2520L laptop with Intel Core i5-5200U CPU, with 8GB installed RAM and Windows 10 Pro operating system.

4.3.1 Inertial Measurement Units (IMU)

For the purpose of this project, a single IMU (Tech IMU V4, Technaid), comprising of a 3D gyroscope, a 3D accelerometer and a 3D magnetometer was mounted on the exterior lateral aspect of the shank, two centimeters below the prominence of tibial tuberosity. A single IMU requires 3.3-4.0V supplied and provides angular velocity in the range +-34.9 rad/s, with a resolution of 0.06 degree/s; acceleration in the range +-39.22 - 156.88 m/s$\hat{2}$, with a resolution of 0.122 **unit??**; magnetic field of +-810 **microTesla** with a resolution of 0.092 microTesla.

The IMU outputs raw data or digital, physical, Quaternion orientation and Direct Cosine Matrix (DCM) orientation formats, with sampling frequencies of up to 100Hz. The parameters of interest that allow detection of gait events are acceleration and angular velocity, therefore 3D Physical acquisition with automatic calibration and sampling frequency of 100Hz was selected. In total,

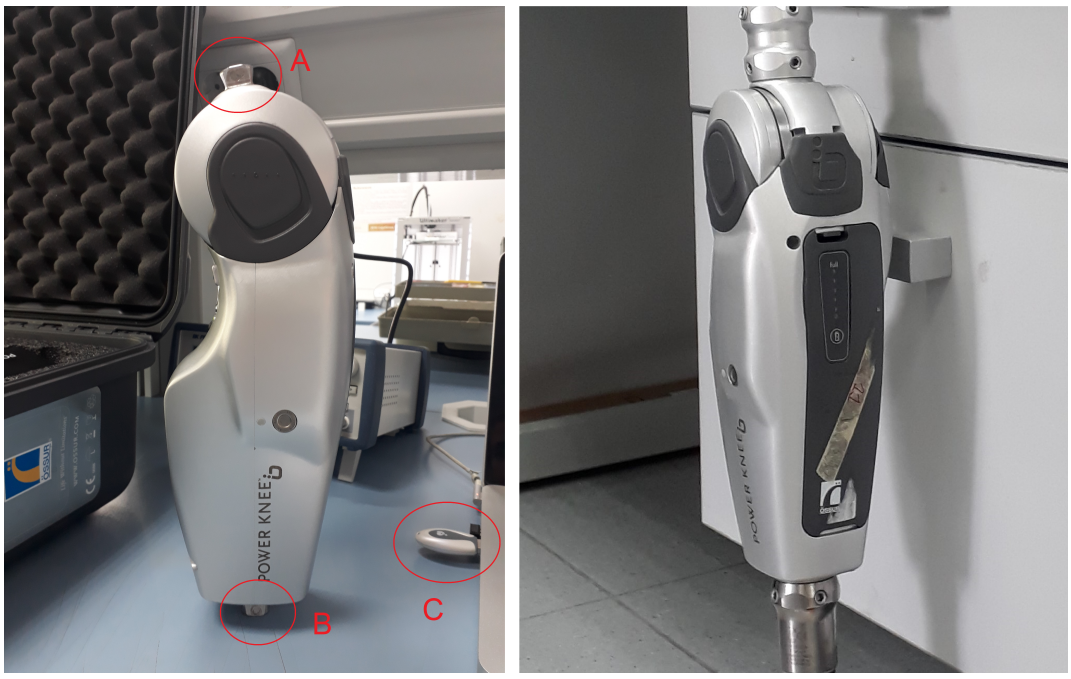


Figure 5: Power Knee II, Össur. A - location of thigh sensors; B - location of shank sensors and load cell; C - dongle for wireless connection to the prosthesis

the IMU transmits ten variables in the following order: acceleration (x, y, z), angular velocity (x, y, z), magnetic field (x, y, z) and temperature.

4.3.2 Force Sensing Resistors (FSR)

Two FSR 406 38mm Square x 83mm (Interlink Electronics) sensors with 0.2N actuation force, sensitivity between 0.2-20N and analogue output were used to capture heel and toe contact. Since the contact thresholds could be determined from the raw sensor data, this was not converted from electrical signal to force. The FSRs output electrical signals through a telephone cable connected to the analogous pins of an Arduino board via two RJ11 ports joined by a Phoenix Contact 1711026 connector, which passes data to the computer. A radiofrequency XBee module is used for wireless communication between the Arduino and the computer The Arduino UNO sensor data reading allows a 115200 baud serial connection. Counting the stop bit, the delimiters and the trigger, 11 bits are used for every 8 bits of data sent. This would normally allow sampling of the sensors at more than 100Hz, which was the desired frequency. However, after data collection it was noticed that the sensors were sampled at only $F_S = 10Hz$, which might be caused by the low processor speed of the computer.

4.4 Data Collection

Data acquisition components are in charge with reading information from the sensors and passing it to the gait event detection algorithm in order to decide on the biofeedback output. The sensors employed for this are one Inertial Measurement Unit (IMU), on whose output the event detection relies, and two Force Sensitive Resistors (FSR, Interlink Electronics FSR 406), which were used for the initial validation of the gait event detection algorithm.

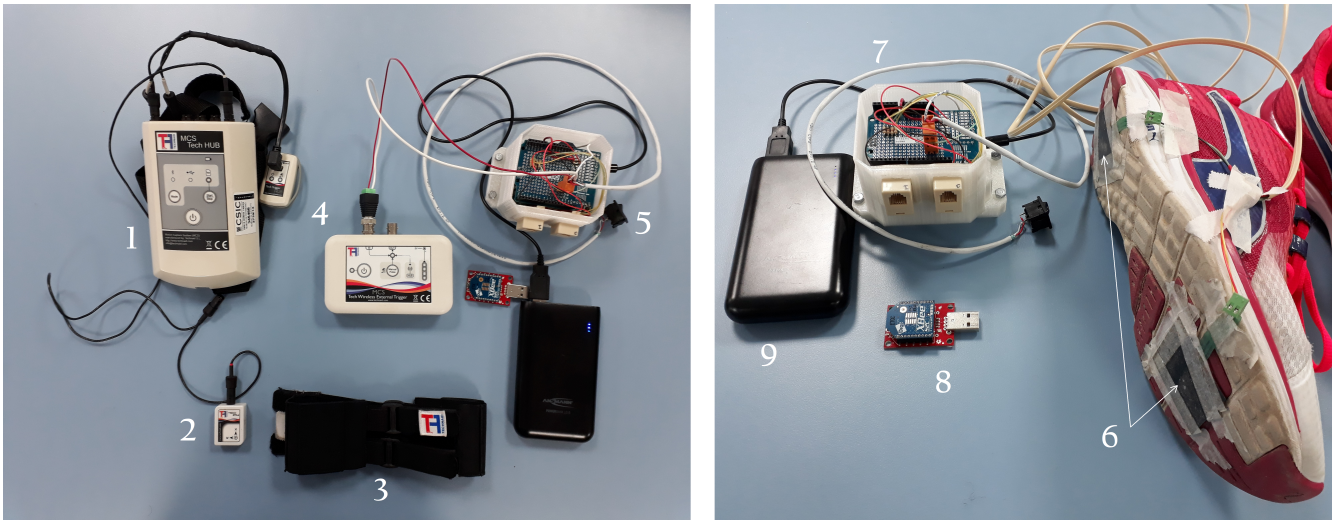


Figure 6: Instrumentation for part I of the software validation: 1-Tech HUB, 2-IMU, 3-elastic strap, 4-trigger, 5-trigger switch, 6-FSR sensors, 7-case containing Arduino UNO and XBee module, 8-USB for Zigbee wireless communication module, 9-power supply.

Inertial sensor data from one IMU was acquired in two ways, first for offline gait analysis and initial validation of the algorithm, followed by collection and analysis of data in real time. In order to maintain data collection and processing as close to that of the parameters from Power Knee II (Össur) embedded sensors, which are sampled 100 times per second, the IMU sensor data was collected at the maximum allowed sampling frequency, $F_s = 100$Hz. Due to the mismatch between the sampling frequencies of the FSRs and the IMU, an external trigger was used to ensure synchronisation of data collection. This was done using the In/Out Trigger Port for Synchronisation (Technaid) connected to the Tech HUB, in the Input mode, and a switch trigger connected to the Arduino board (Figure 6, 5). When the switch is pressed, a series of pulses is sent to the trigger, which in turn sends the ready/start/stop commands to the Tech HUB; for FSR triggering, when the switch is on, it sends a series of 1's to the Arduino, allowing identification of the start and end points in the output file.

Data collection from the FSRs is performed through wireless connection to the Simulink model initially developed for Functional Electrical Stimulation control [72]. The system consists of an Arduino UNO board equipped with a battery connected to the two FSR sensors that are attached on the sole of a shoe, with the information being sent through Zigbee communication. The Arduino board and the XBee radio frequency module are encased in a 3D printed box which can be attached to the waist of the subject. The FSR sensors are connected to the Arduino board through telephonic cables. The data collected is in analogous form in order to allow threshold-based detection of foot contact. The pressure detection interval begins at 0 and saturates at 1023.

4.4.1 IMU Communication interfaces

Tech MCS

The Tech MCS interface is the default data acquisition means provided by Technaid to capture and export data for offline analysis. It requires the use of the Tech Hub, which enables simultaneous

recording from up to 16 IMUs connected to the four ports. The hub provides power to the IMUs and can be attached on the subject's waist to allow free movement and recording data in offline PC mode, on a MicroSD card. The SD card set-up specifications include the type of data to be recorded (Physical, Digital, Quaternion), the sampling frequency (1-100Hz), and labels of the IMUs, for multiple IMU **setups**. The data thus recorded was exported in *.cpp* format, for which a Matlab code was written to extract the relevant information for analysis and processing.

Figure 7: TechMCS interface containing plots of collected data, name of the project, name and number of the IMUs. In USB recording mode, this allows visualisation of data in real time.

CAN

For real time data analysis, a CAN (Controller Area Network) communication protocol was required. CAN is a highly integrated serial bus system destined to be used for communication with smart devices and is the standard network used for vehicles [73].

Obtaining data from the IMU through the CAN bus involved the following steps, expanded in the flowchart in Figure 12:

1. physical connection of the IMU-CAN to the PC through USB
2. obtaining the identifier of the IMU used, through PCAN View software.
3. sending a one-off message that informs the CAN bus that Physical data will be collected
4. sending periodic *pooling* command (empty message) that asks the sensor for the captured data
5. receiving the data in 32bits float format

Sending and receiving messages through CAN bus was first done in PCAN View software, a CAN monitor for viewing, transmitting, and recording CAN data traffic [75], which relays the data in

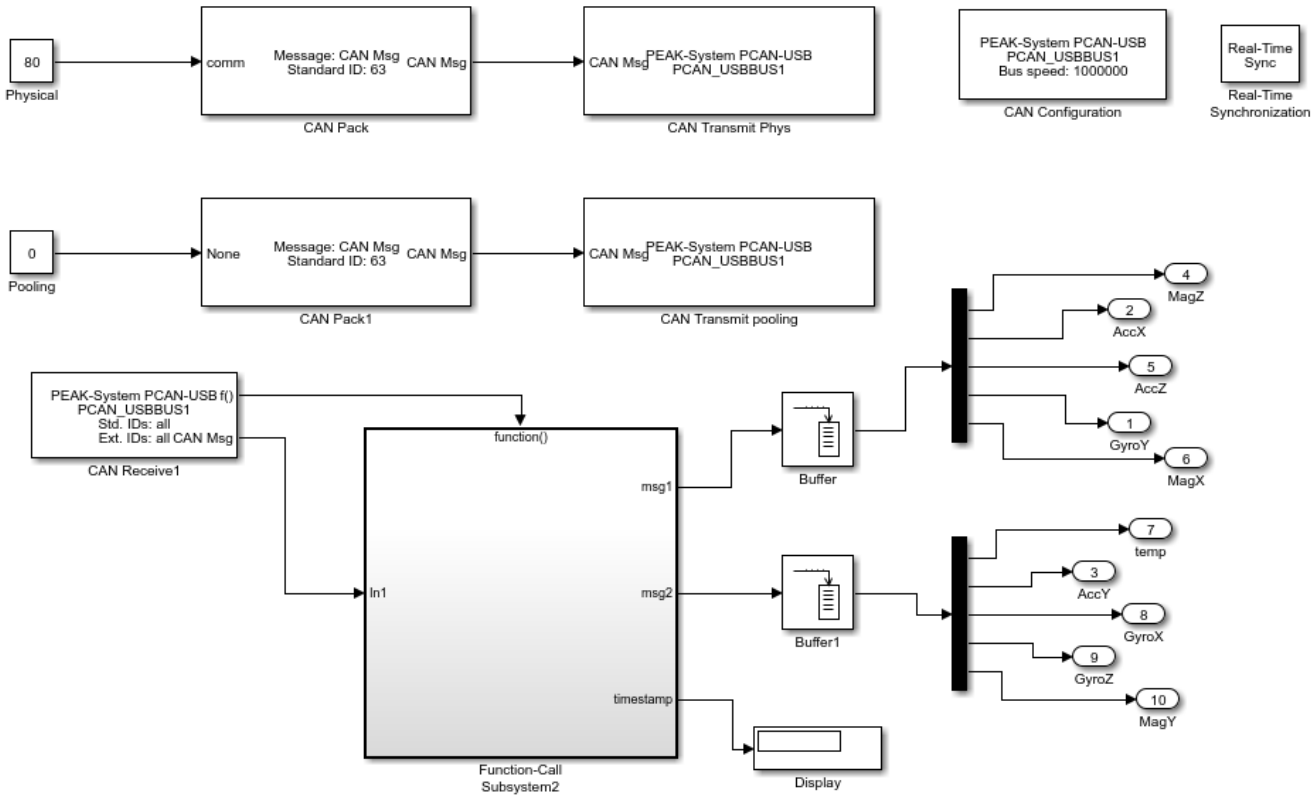


Figure 8: Configuration of PCAN USB, message transmission and receive to the IMU. The Function-call Subsystem contains the data Unpack block (Figure 9).

decimal or hexadecimal format (Figure 10). However, for real time gait event detection, the data had to be analysed as it arrived to the PC and integrated with the graphical user interface and the audio feedback developed in Simulink (Mathworks, R2017a); for this reason, a Simulink CAN connection was opted for, using the additional toolbox *Vehicle Network Toolbox* (Mathworks). A bug fix was necessary for version R2017a of Matlab [76] due to the run-time error *Unable to query hardware information for the selected CAN channel object. Struct contents reference from a non-struct array object.*

Through the Simulink model presented in Figure 8, the messages were packed and sent via respective blocks and the data from IMU was received and unpacked taking into account the size and format specified in the TechnAid User Manual [74].

The CAN Configuration block establishes the communication parameters: the device (PEAK system PCAN USB) and the communication speed (1,000,000 bps). The message is transmitted in decimal values as data type uint8 in a Constant block, sent to the CAN Pack block, that specifies the ID of the IMU that will receive data, and to the Transmit block, where periodic sending can be chosen. The first message - physical - is sent only once as the decimal value 80; pooling is an empty message, therefore the value 0 was sent to the IMU at a period of 0.01 seconds (Figure 8). The data is received through the Receive block, and unpacked inside a Function-call Subsystem, as shown in Figure 8, to ensure unpacking is done every time data arrives. An Unpack block can translate up to 8 bytes of data, while each of the ten physical values from the IMU is encoded in 32 bits (4 bytes). The unpacking of the data and block configuration is presented in Figure 9.

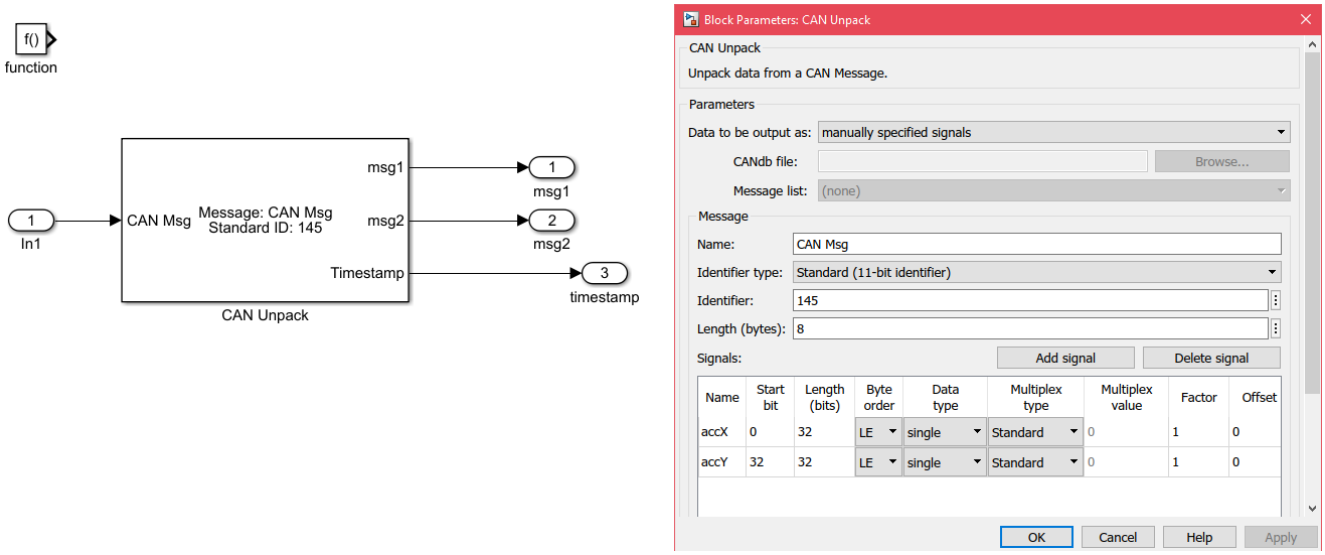


Figure 9: Function-triggered data unpacking (left) and configuration of Unpack block to manually receive two messages of type *single* at each unpacking stage.

Table 1: Data format as received through CAN communication from an IMU.

CAN message	Output 1	Output 2
Message Structure	AccX, AccZ, GyroY, MagnetX, MagnetZ	AccY, GyroX, MagnetY, Temp
Message Length	XXXX, XXXX, XXXX, XXXX	XXXX, XXXX, XXXX, XXXX

A complete sensor reading requires five packets of data to arrive. Figure 10 shows the 40 bytes of data and the order in which they arrive, as recorded with PCAN-View for Windows (PEAK Systems). The output of the Unpack block was stored in a buffer of size 5 with zero overlap so as to allow demultiplexing of the data (Demux block). As a result, the output data could be sorted into its respective sensor type. Inspection of the data revealed that the delays **incurred** during message transmission and receiving caused an offset of one data packet - hence the first data set is Z-axis magnetometer and temperature, respectively (outputs in Figure 8).

```
;   Start time: 30.11.2018 10:49:00.757.0
;   Generated by PCAN-View v4.2.1.533
;
;   Message Number
;   |         Time Offset (ms)
;   |         |        Type
;   |         |        |        ID (hex)
;   |         |        |        |     Data Length
;   |         |        |        |     |   Data Bytes (hex) ...
;   |         |        |        |     |   |
;---+--   ----+----  --+--  ----+---  +   -+ -- -- -- -- -- -- --
     1)      1509.9  Tx         003F  1   50
     2)     16175.5  Tx         003F  0
     3)     16176.0  Rx         0093  8   E7 57 68 40 FE 06 E6 40
     4)     16176.1  Rx         0093  8   05 E2 B4 C0 B9 E7 12 3B
     5)     16176.2  Rx         0093  8   AC 37 85 3B 4C 39 5C BB
     6)     16176.3  Rx         0093  8   63 6C 81 42 A0 A3 69 C2
     7)     16176.4  Rx         0093  8   82 43 56 42 88 87 03 42
     8)     16276.3  Tx         003F  0
     9)     16277.2  Rx         0093  8   A2 5B 6C 40 31 7B E6 40
    10)     16277.3  Rx         0093  8   6B 17 B4 C0 8D 8A 8F 3B
    11)     16277.4  Rx         0093  8   B9 15 65 3A 1B F0 19 BB
    12)     16277.5  Rx         0093  8   73 6D 81 42 10 50 69 C2
    13)     16277.6  Rx         0093  8   AC 9C 55 42 E8 E7 03 42
    14)     16376.2  Tx         003F  0
```

Physical command

AccX

AccY

GyroY

Pooling command

Figure 10: Snippet of data acquired from one IMU using CAN communication and PCAN-View (PEAK Systems) software for trace recording. The first transmitted message is the "Physical" command, followed by "Pooling", with periodic transmission every 0.01s. Each received message contains data from two sensors - first and second group of 4 bytes, as detailed in Table 1.

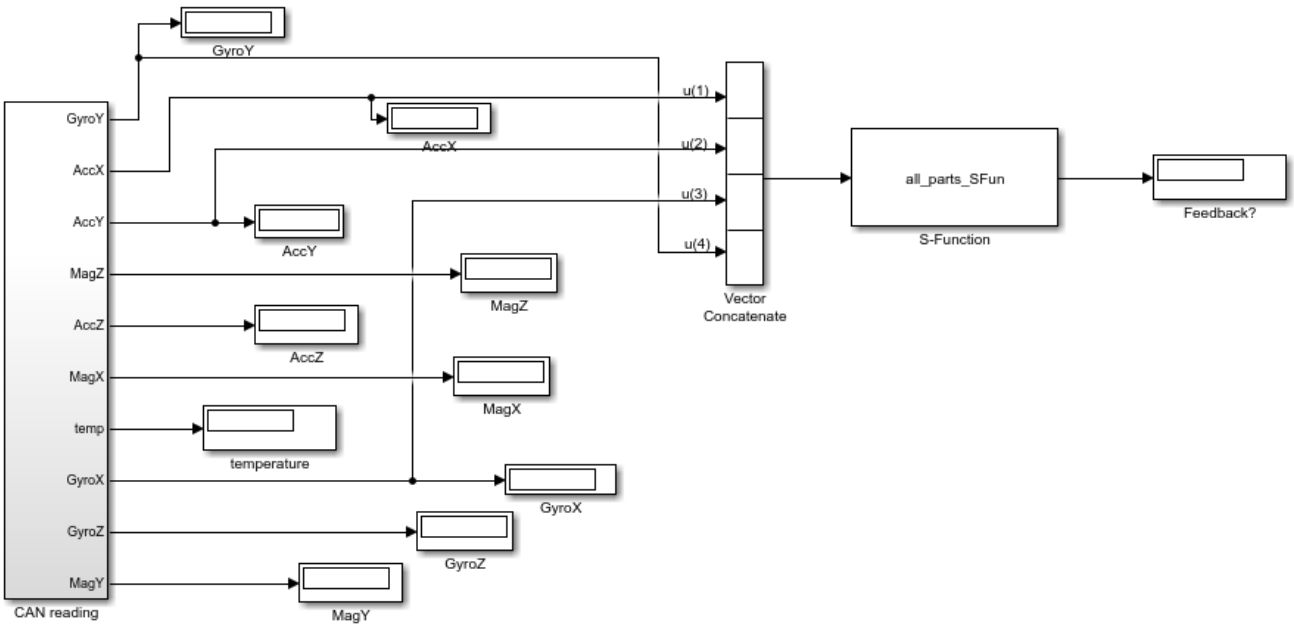


Figure 11: Output sensor data from IMU relayed to the S-function for gait analysis.

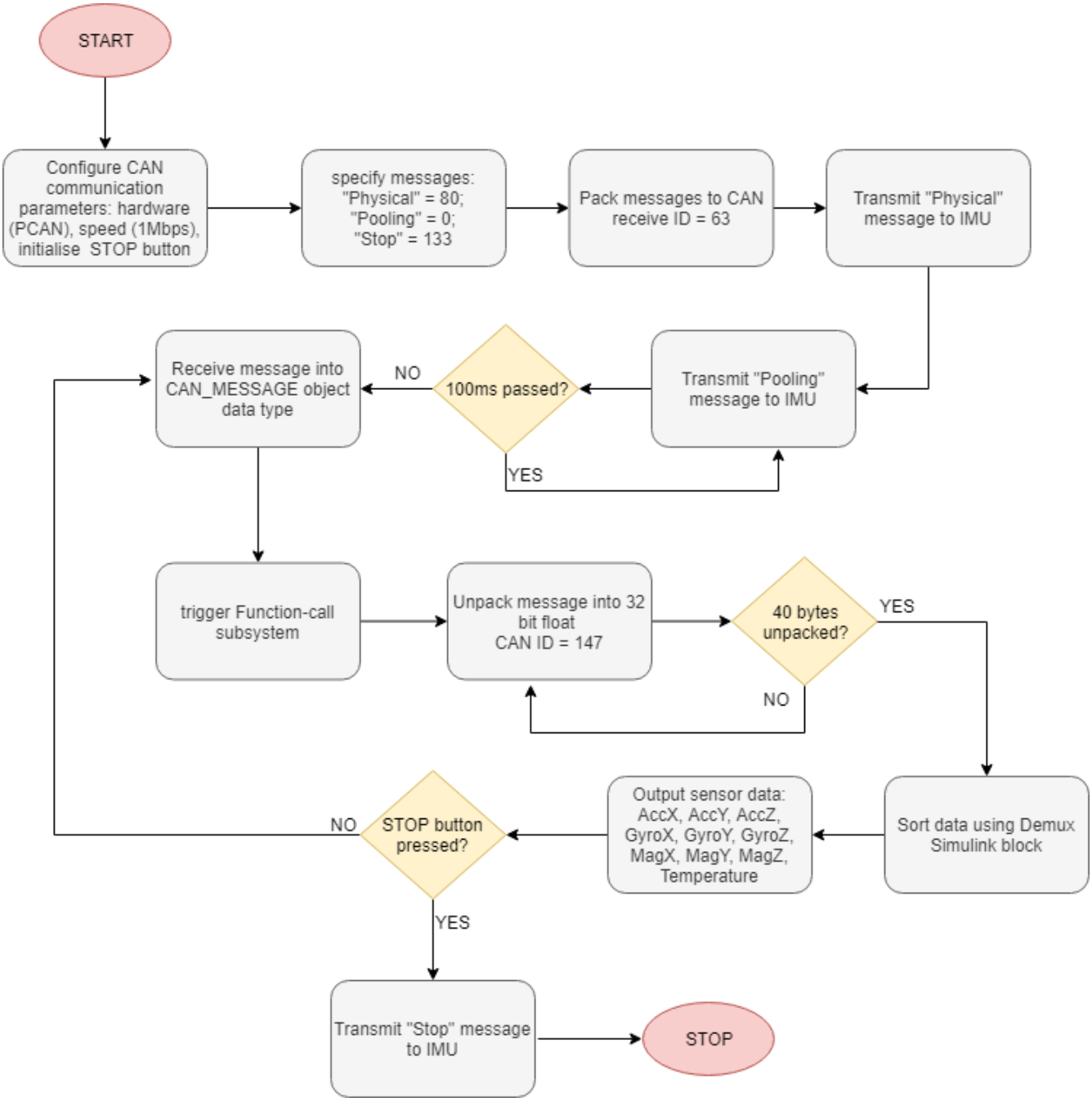


Figure 12: Flow chart of data acquisition through Simulink using CAN protocol

4.5 FSR communication

Data from the two FSRs is collected via the Arduino in Simulink. The Serial Receive block was configured in the Serial Configuration block to receive 8 bit long data through COM port 8 at 115200 baud rate in Little Endian byte order, specifying the final character to be "\n". The message is received in ASCII code, therefore 48 is subtracted from each bit to obtain the decimal values.

Each message contains 11 bits which are separated using the Demux block. The message is then reconstructed into the two sensor values, two separators and a trigger. The sensor output is formed using Equations 1 and 2 below [72]:

$$SensorMTP = bit3 * 1000 + bit4 * 100 + bit5 * 10 + bit6 \tag{1}$$

$$SensorHEEL = bit8 * 1000 + bit9 * 100 + bit10 * 10 + bit11 \tag{2}$$

The trigger (first bit) and the two sensor outputs are then sent to Matlab workspace using To Workspace block.

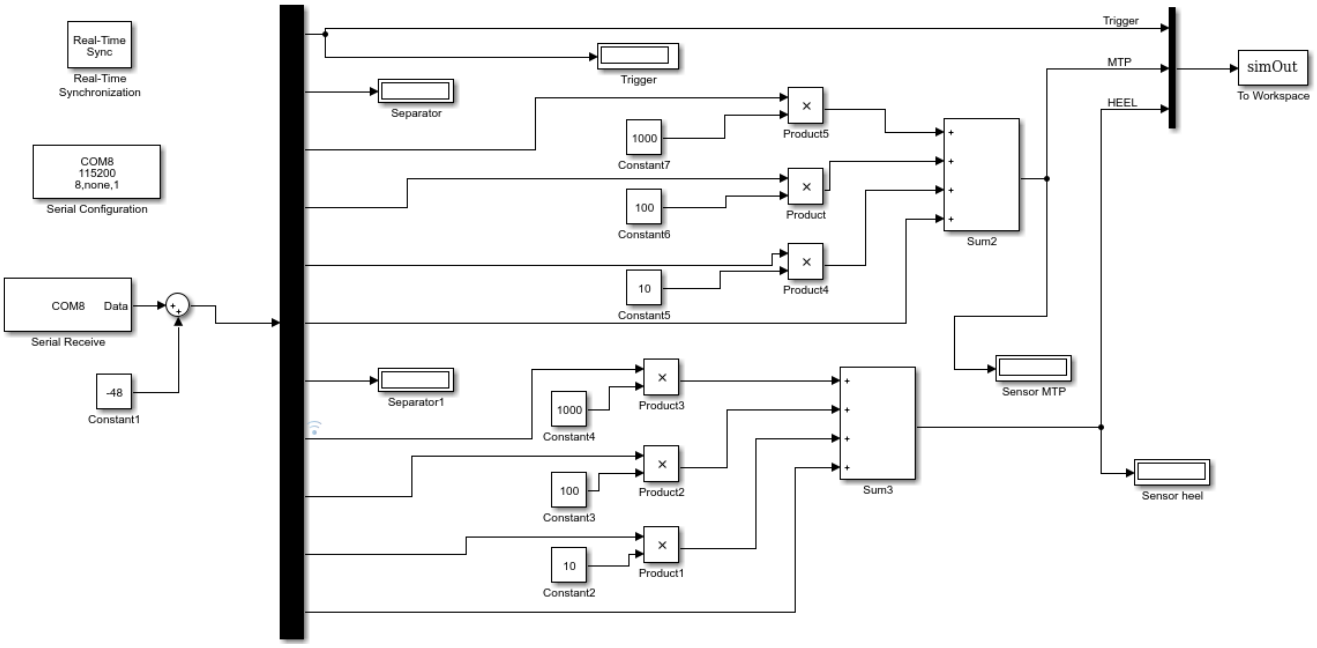


Figure 13: FSR data collection model in Simulink, adapted from [72]. 11 bits of data received through Arduino are sorted using Demux block. The messages are reconstructed to form the two sensor outputs and sent to Matlab workspace for offline signal processing.

The trigger is used to synchronise inertial sensor data with force sensor values. As mentioned in Section 4.7.3, the trigger field outputs a string of zeros when the switch is off; when pressing the switch, the output becomes one and the Tech HUB begins recording on the MicroSD card. To stop the recording, the switch is pressed once more, changing again the trigger output from zero to one.

4.6 Gait event detection

Offline visual inspection of data from the IMU worn by a subject walking on flat ground at varying speed has shown stable output of both gyroscope and acceleration in the three dimensions (Figure 14). Filtering for noise removal is [presented] in literature with wide variations, with gyroscope signal being low pass filtered using cut off frequencies between 3 and 31Hz [79, 35, 95]. Moreover, pre-filtering of flat ground walking data did not indicate significant improvements in waveform quality, meaning that gait event detection could be performed in real time, on raw physical signals. This, together with conclusions from literature review [84, 78], enabled further analysis to determine the signals to be employed for detection of the four gait events.

Studies documenting gait event detection with a single axis gyroscope mounted on the foot or shank to measure angular velocity in the sagittal plane presented high accuracy and robustness of the systems [80, 84], which was confirmed by performing offline detection trials of the main gait events on data from two subjects: one healthy person and one robotic prosthesis user walking with Power Knee II (Össur). Offline detection of the main gait events (IC, TO, MSW) enabled segmentation of gait cycle to perform analysis on the average duration of stance phase, as well as the minimum and maximum thresholds. Therefore, detection of the gait events was initially based solely on the angular velocity of the shank in the sagittal plane.

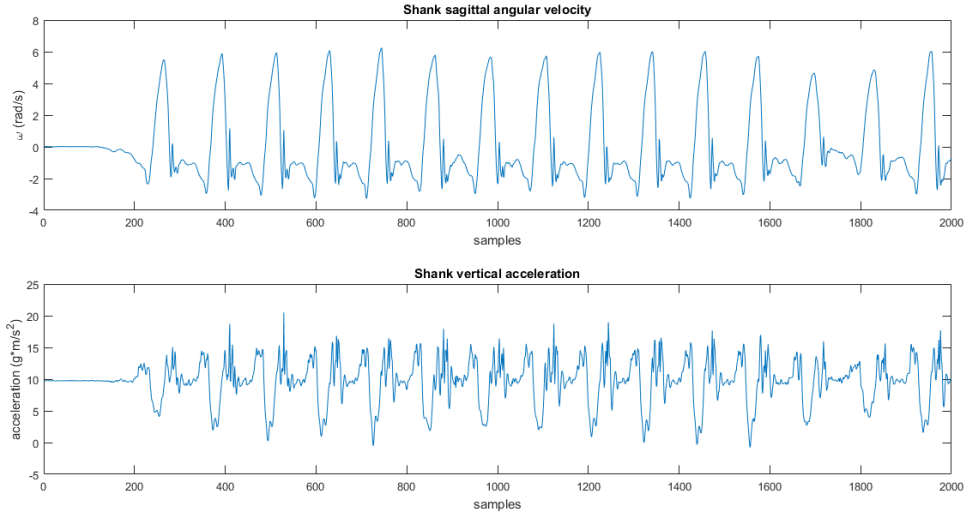


Figure 14: Shank sagittal angular velocity and vertical acceleration

Given the axes and planes conventions (Figure 2), initial contact of the leg to the ground is identifiable as a minimum (trough) in sagittal angular velocity, occurring just after the maximum of the gait cycle, which corresponds to mid-swing. At the end of the swing phase, the knee extends, the foot dorsiflexes and increases velocity, and the shank follows a clockwise trajectory which peaks (negatively) at the first contact. While the foot follows the roll-over shape, during stance phase, the absolute value of the angular velocity is low. During the final segment of stance phase, after the heel is off the ground, the body CoG is propelled forward and the shank moves clockwise with respect to the ankle, using foot plantar flexion, thus Toe Off event is found at the second minimum of the gait cycle.

4.6.1 Detection and feedback algorithm

The flow chart in Figure 15 is a simplified representation of the algorithm that provides a graphical user interface, detects the above mentioned gait events, computes the average stride time of the subject and provides audio feedback when necessary. The detection algorithm mainly makes use of the sagittal gyroscope data and follows work presented in [84]. However, to increase its robustness and allow for extended functionality to be added, the input to the algorithm consists of four of the ten variables output by the IMU: vertical and sagittal acceleration and vertical and sagittal angular velocity. The variables are fed into a Matlab S-function (Matlab R2017a), a function that enables the detection algorithm and Graphical User Interface (GUI) to be added to the Simulink model as a custom-built block [83].

The first condition that allows further gait events to be identified is the detection of a maximum point in the sagittal angular velocity, corresponding to MSW. The condition for identifying the initial contact is comprised of two parts:

1. check that the point is a local minimum using point derivatives
2. eliminate potential false detection by restricting the minimum difference of consecutive IC occurrences to a half of the current stride duration

Due to the fact that, at IC, the angular velocity in the sagittal plane can roughly vary between -0.5 and -2 rad/s, it is difficult to define a minimum threshold. Instead, the IC detection was improved by including the condition that the Y-axis gyroscope local minimum has to be in proximity to the local maximum in sagittal acceleration.

For the detection of TO and MSW, the minimum and maximum angular velocity adaptive thresholds were used respectively, together with the local extrema condition similar to that of IC. In addition, the TO event, which was subject to notably more false detections, was confined to occur after 40% of the gait cycle has passed.
The real time update of detection thresholds and stride time were employed in order to account for inter-step variations for the same subject and to eliminate the need for calibration. The first three steps of the subject can be considered as calibration of the aforementioned values.

4.6.2 Graphical User Interface

The graphical user interface was constructed with the aim of providing a visual indication of the correct functioning of the detection and feedback algorithm and was adapted from a GUI employed in parameter modulation for functional electrical stimulation [72]. The interface comprises four sections: *Sensors*, *Thresholds*, *Warnings or Events*, and *Visualisation* of sensor data. The first section updates the values collected from the vertical acceleration and sagittal angular velocity sensors, collected with a sampling frequency of 100Hz, once the *Start* button is pressed and until the *Pause* or *Stop* buttons are pressed. The Thresholds section shows the computed average stride time, minimum and maximum angular velocity thresholds in the current step, which are used to detect TO and MSW events respectively, as well as maximum vertical acceleration threshold, used to identify IC. The Warnings/Events field displays the current gait phase and informs when the user has started or stopped walking. In the case of increased stance phase length, a message is displayed. The two plots in the Visualisation section display in real time 200 data points from

the sagittal angular velocity and vertical acceleration sensors, corresponding to two seconds. The bottom section of the GUI contains a brief description of the algorithm, the authors of the GUI and the logos of the sending and receiving institutions. The detection begins when the *Start* button is pressed and can be paused by pressing the same button again. Pressing the *Stop* button stops the detection and closes the GUI. The *Save* button allows saving of data when the detection is paused. The data is saved automatically when the GUI is stopped.

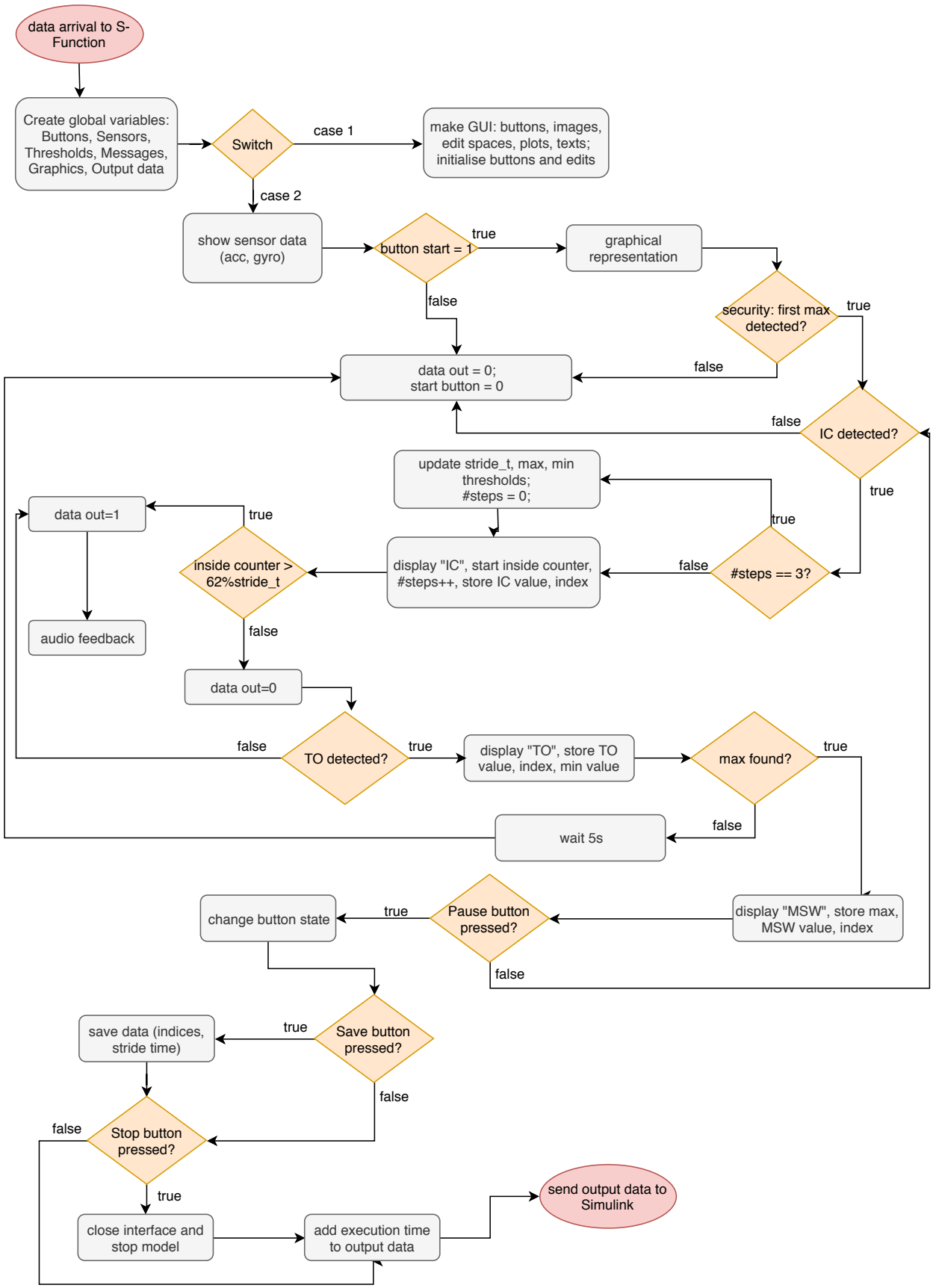


Figure 15: Simplified flow chart of the detection and audio feedback algorithm. IC = initial contact; TO = toe off; MSW = mid-swing; #steps = counter that updates the thresholds and stride time after every three steps.

4.7 Experimental protocol

The gait event detection and biofeedback tool has been tested for validation on ten healthy subjects. The experimental protocol consisted of flat ground walking at the subject's most comfortable speed and treadmill walking at set speed trials, which, as seen from the Eurobench literature review of lower limb prostheses, are the two most assessed locomotion types for feasibility studies and prototype testing. Walking on flat ground was preferred because it enables testing of the gait event detection algorithm in a basic real life situation, which introduces more gait parameter variation across the trial, thus offering more opportunities to assess the robustness of the system. Allowing the subjects to walk freely around the laboratory corridors, which included 180°direction changes, stopping to open and close doors, and turning corners provided simple "white box" algorithm testing conditions. Treadmill walking was used due to space constraints imposed by the real time hardware system employed in the second part of the study. The data analysis and audio feedback were performed on the laptop, the GUI was displayed on the laptop screen and, when applicable, the audio signal was transmitted through the laptop speakers.

For the first part of the study, the subjects were equipped with a wireless data collection system consisting of one IMU and two FSRs. The second part involved walking on the treadmill while wearing one IMU that was wired to the laptop.

4.7.1 Subjects

The ten subjects (5 males, 5 females) who participated in this study have an average age of 26.3±3.4 years, average weight of 69.3±111.8kg and average height of 171.6±110.05cm. Detailed information on individual subjects is found in Table 2.

Inclusion/Exclusion criteria
In order to be included in the study, the subjects had to be healthy, with no limb amputation, no visible gait asymmetry and no history of spinal cord injuries, or other conditions that could affect their gait.

The exclusion criterion was the size of the foot, because the FSR sensors have to be placed in a location that captures data from the heel and ball of the foot. One subject was excluded from the study for this reason, because their feet were too small and the heel and MTP position changed while walking.

4.7.2 Receiving the subject

The subject was given an information sheet where the two parts of the experimental protocol and their aim were explained and it was emphasised that they could stop the procedure at any moment and withdraw from the study, without the need to provide any reasons for doing so. Written consent was then obtained and their age, height and weight were collected. Permission was obtained from the subject for photos and videos to be taken during the experiments.

4.7.3 Set up

It was decided that the IMU would be placed on the external lateral side of the shank, such that the IMU X axis corresponds to the vertical axis of the body, pointing upwards, the Y axis captures

Table 2: Age, height and weight of the ten subjects who participated in the study.

Subject #	**Gender**	**Age**	**Height [cm]**	**Weight [kg]**
1	M	24	176	71
2	F	33	175	70
3	F	26	154	57
4	M	23	179	94
5	M	25	171	75
6	F	23	170	62
7	M	25	187	82
8	F	31	165	59
9	M	25	180	64
10	F	28	159	59

information in the sagittal plane and the Z axis is aligned with the frontal axis of the body. The subject was asked to sit on a chair to allow identification of the chosen landmark, the head of the fibula. The IMU was attached to the right shank of the subject, 2cm below the landmark, using elastic straps with pockets (Technaid). Two sets of two FSRs (FSR 406, Interlink Electronics) had been previously placed on the soles of a shoe of 24.6cm and on a custom made sandal of length 27.3cm, using paper tape. The resistance of the FSR drops as pressure is applied to its active surface, therefore the position on the shoe soles is key to correctly determine foot contact. The location of the FSRs where the applied pressure was highest had already been investigated [72] using the instruments presented in Figure 6, as such it was decided that the FSRs should be placed on the heel (HEEL FSR) and at the metatarsophalangeal (MTP FSR) joints. It was noticed that, when placing the sensors on the insoles of the shoes, values were registered although the foot was off the ground, thus it was decided to place the FSRs on the outside of the shoe, using paper tape. Only healthy subjects were used in this study, therefore it was expected that the pattern of the pressure exhibited on the heel and forefoot would not have significant inter-subject variability. The variation in subjects' feet length was accounted for by using two pairs of shoes (Figure 17) and by moving the MTP FSR accordingly.

These were moved to coincide with each subject's heel centre and first metatarsophalangeal (MTP) joint, in order to ensure optimal data collection. One of the two pairs of shoes was given to the subject according to their shoe size. The IMU sensor was always placed on the right leg to coincide with the side of the FSRs which were fixed to the soles of the right shoe.

The first part of the algorithm testing involved offline collection of data from the subject walking on flat ground for two minutes, therefore the Tech HUB (Technaid) was attached to their waist using an adjustable strap (Technaid). The IMU was connected to the Hub via cables with binder connectors, as specified in Technaid's *"Recommended setup for a 16 IMUs network on the human body"* [92], using a cable of type 3 and a cable type 5 (Technaid) connected to port 1 on the Hub (Figure 16). The synchronisation of the data collection system was achieved using the external trigger, which was wired to the Arduino box. The subject was given the choice of carrying the FSR power source, the Arduino box and the trigger remote control in a backpack.

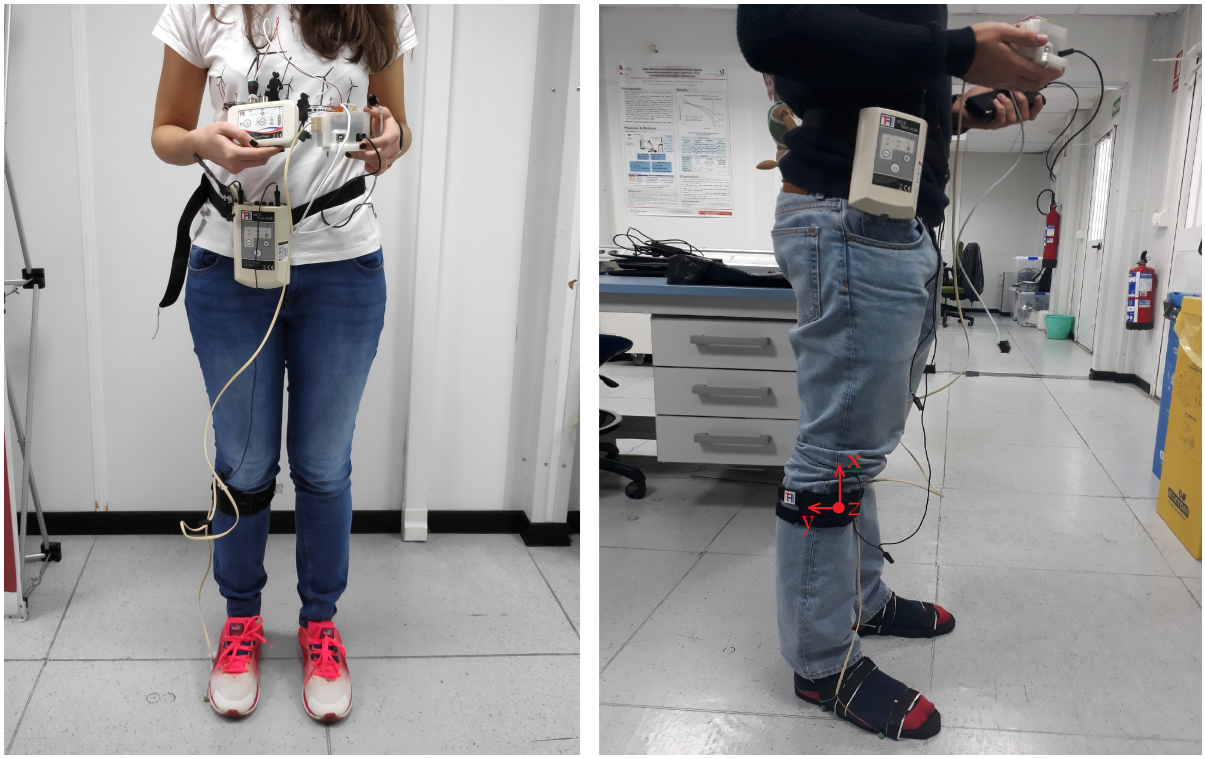


Figure 17: Setup comprised of IMU strapped on the right shank (inside pocket), Tech HUB, two FSRs on the soles of the right shoe connected through a telephone cable to the Arduino UNO box, the trigger and the power supply (not visible). Left - subject wearing shoes for sizes 38-40. Right - custom made sandals [72] for sizes 41-44.

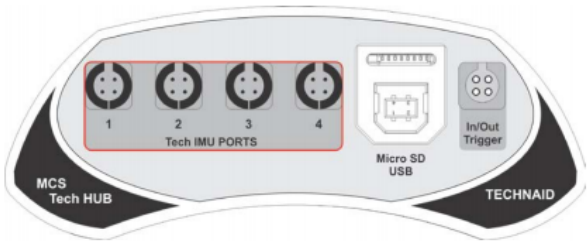


Figure 16: Tech HUB, Technaid: four IMU ports that receive up to 16 IMUs, MicroSD slot, USB cable slot for connection to the PC, and the Trigger port.

For the second part of the feasibility study, which involved real time testing of the algorithm, the CAN communication replaced offline recording using TechMCS. The same ten subjects performed a fourty second walking trial on a treadmill (Domyos TC450, Decathlon), due to the fact that the Peak CAN was connected via USB to the laptop. Hardware limitations have restricted the walking trials to 40s - when testing for longer than a minute, system latency increased on the laptop and the results could not be retrieved. An IMU (v.4 Technaid) was fixed on the same location on the right shank, using the elastic strap. This was connected to the CAN unit through a coaxial cable that allows simultaneous recording from two IMUs (Figure 19).

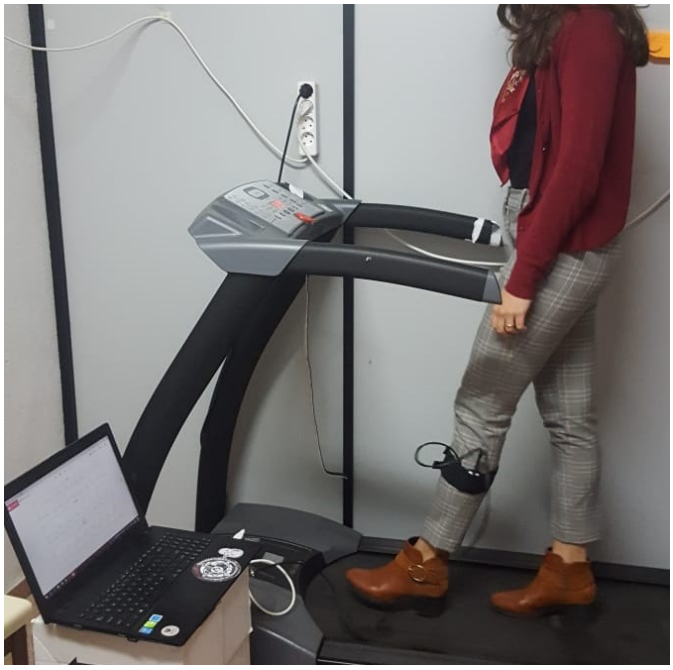


Figure 18: Setup for the real time testing of the gait event detection and feedback tool

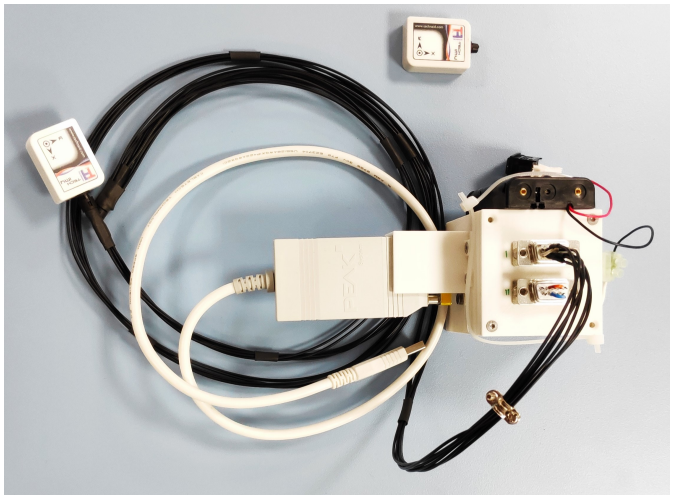


Figure 19: Instruments employed for part II of the software validation- real time testing using CAN communication protocol: IMU (Technaid), coaxial cable with two binder connectors, PCAN-USB (Peak System), 3x1.5V accumulators, CAN communication circuit.

4.7.4 Calibration

Motion capture in Physical mode with the IMU does not require manual calibration, as this is done automatically [74]. The FSRs' analogue output begins at 0 - no contact and saturates at 1023. Various walking trials, including one legged stance, two legged stance, standing on the toes and heel stance, were performed in order to determine the threshold values for foot contact and the maximum value for each sensors. The empirically determined values can be found in Table **??**.

4.8 Data Analysis

The gait event detection system makes use of signals recorded from a single limb, therefore the results are presented in terms of stride, or gait cycle, found from consecutive ICs of the same limb, rather than steps.

4.8.1 Simulation of real time detection

For the simulation of the real time detection, the data recorded from the IMU-FSR system was visually inspected in order to determine the total number of strides from each subject. This number was found using the peaks in sagittal angular velocity obtained from the IMU acquisition, corresponding to the MSW. Additionally, an offline detection algorithm was written in Matlab to verify the total step number for each subject (Appendix, 10.2), the location and the value of each event (Figure 26). These were subsequently used to compute the accuracy of the detection. The output of the Simulink model comprises the values and locations of the three gait events, as detected by the real time algorithm.

The data collected as per Section 4.4.1 was used for the first part of the study, to evaluate the accuracy of the gait event detection in simulated real time.

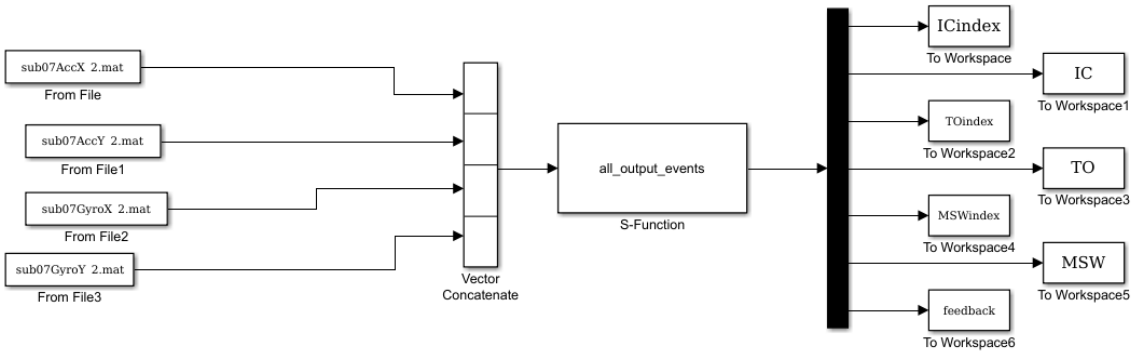


Figure 20: Simulink model used to simulate real time gait event detection and feedback.

The sagittal and vertical angular velocity and the sagittal and vertical accelerations of the shank were concatenated into a vector (Figure 20) before arriving at the S-Function, which ran as detailed in the flowchart in Figure 15. The model outputs seven variables, namely the value and location of the IC, TO, MSW, and "feedback" - a string that takes the value zero when weight shifting is performed correctly, and one when audio feedback is necessary. This is then used as a switch to

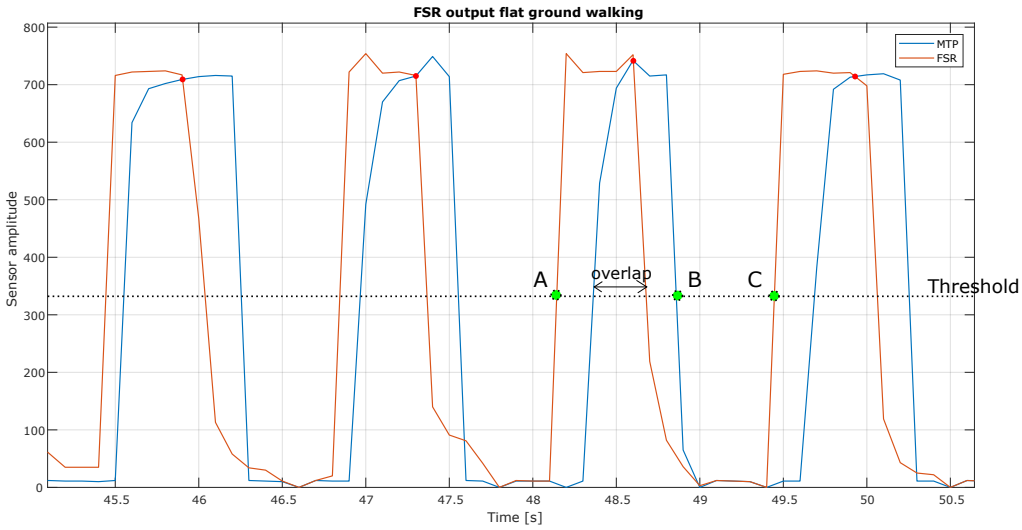


Figure 21: FSR output showing four steps, the threshold used to determine if the heel or ball of the foot are on the ground, and the points used to compute the stance percentage.

turn the sound block on.

The FSR data acquired was used to cross-validate the information from the two sensor systems (IMU, FSR). A function that finds the intersection point between two curves was applied to find the initiation of foot flat (FF). The FSR system is comprised of the heel and MTP join sensors, therefore the output from the two sensors was checked to find: a) the total number of steps recorded using FSR, from the intersection points shown in red in Figure 21; b) the potential of the data to provide information that complements the inertial sensor; c) the stance time as percentage of stride time.

When the value of the sensor crosses the threshold, established empirically at 320, it is considered that the foot segment is on the ground. Thus, in Figure 21, point A corresponds to IC (heel touches the ground), point B is TO, when the MTP output falls below the threshold, and point C is the following IC. The intersection point shown in red corresponds to FF. The equations below were used to compute stride time, stance time and the overlap (FF).:

$$stridetime = (first\ sample\ SensorHEEL\ current) - (first\ sample\ SensorHEEL\ previous) \tag{3}$$

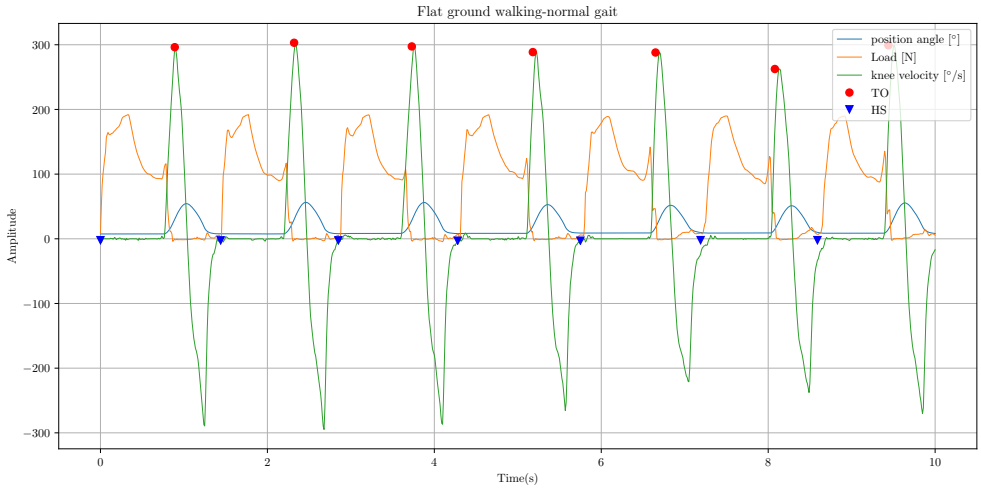
$$strancetime = last\ sample\ SensorMTP\ -\ first\ sample\ SensorHEEL \tag{4}$$

Where *first sample* and *last sample* are the first and the last value recorded above the threshold, for one step. Ideally, these would be located close to points A and B marked on the plot.
The feedback timing assessment would have been done by checking the Simulink output ("feedback") against the computed stride time to determine whether the audio signal was given when necessary.

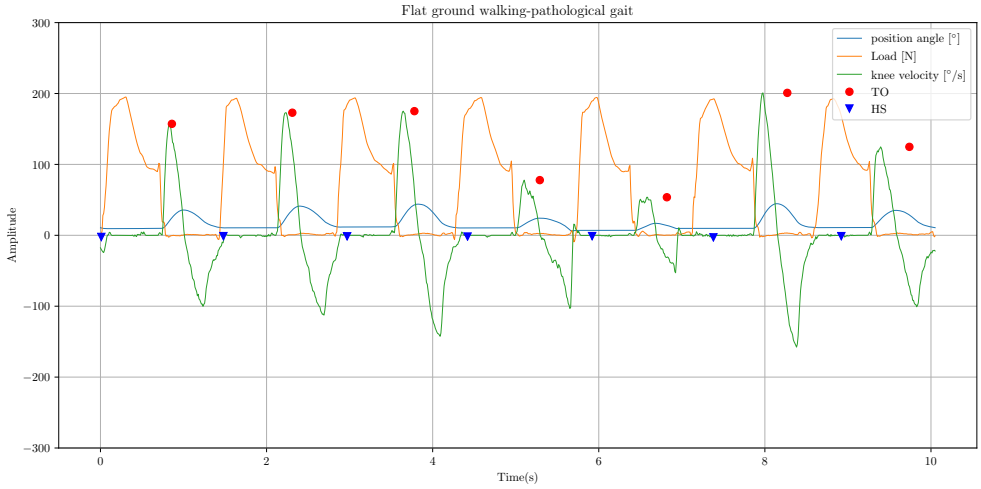
5 Results

5.1 Prosthesis walking trials

The data collected from the experimented prosthesis user was analysed as detailed in Section **??**. The three signals used for gait analysis were represented on the same plot in order to show the detection accuracy with respect to the true events, and how these influence the calculation of stance time as percentage of stride time.



(a) Power Knee angular velocity, position angle and load in normal gait



(b) Power Knee angular velocity, position angle and load in pathological gait

Figure 22: 10 second recordings used to identify gait features. Toe off and heel strike are marked.

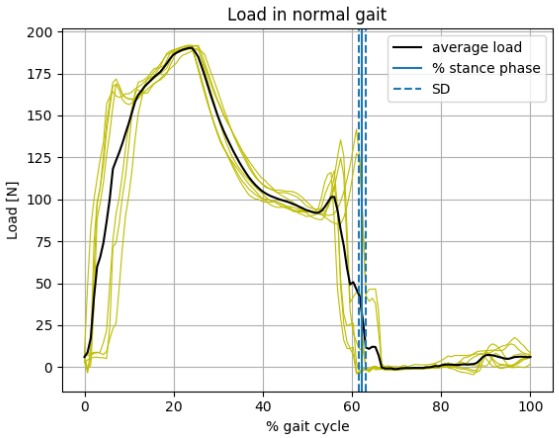
The initial contact (heel strike) and toe off events were used to compute the average of the maximum knee angular velocity and percentage of stance phase in a stride.

Table 3: Calculated values of the mean maximum angular velocity over the seven steps and the percentage of stance phase out of a gait cycle, for the Power Knee limb.

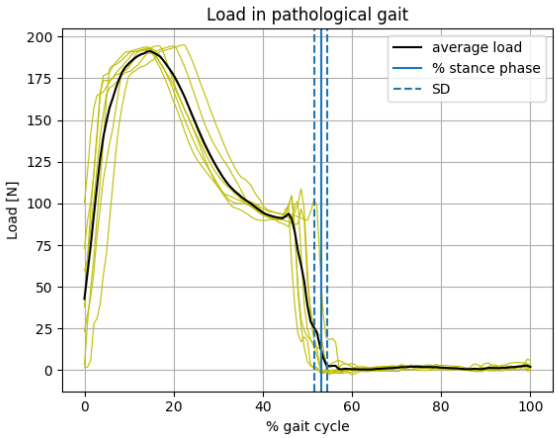
	average max velocity(rad/s)		% gait in stance phase	
	velocity	SD	% stance	SD
normal gait	5.07	0.22	62.2	0.8
pathological	2.40	0.88	53.0	0.87

5.1.1 Load cell data

The knee loading curves obtained from the embedded load cell during flat ground walking in a) normal walking pattern and b) asymmetrical gait are plotted in Figure 23, based on the segmentation performed using the detection of the initial contact, Figure 22. For the case of late swing initiation, the gait segmentation was modified after visual inspection, due to variation in stride time, to ensure that the calculated stance percentage reflected any deviation from the 62-38 proportion. The average and standard deviation of stance phase as percentage of a stride is represented from Table 3.



(a) Normal load per stride, average load and SD. Weight shifting on average occurring at 62% gait cycle.



(b) Load per stride in late swing initiation, average load and SD. Weight shifting is performed on average at 53% gait cycle.

Figure 23: Load curve in one stride and average load calculated over seven strides. The values are obtained from Power Knee embedded load cell.

5.2 Detection and feedback algorithm

5.2.1 Interface

The interface described in Section 4.6.2 is presented in Figure 24 below, with the most recent event detected being the initial contact, called "Heel strike" in the interface. The values of the sensors correspond to the last point represented on the plots.

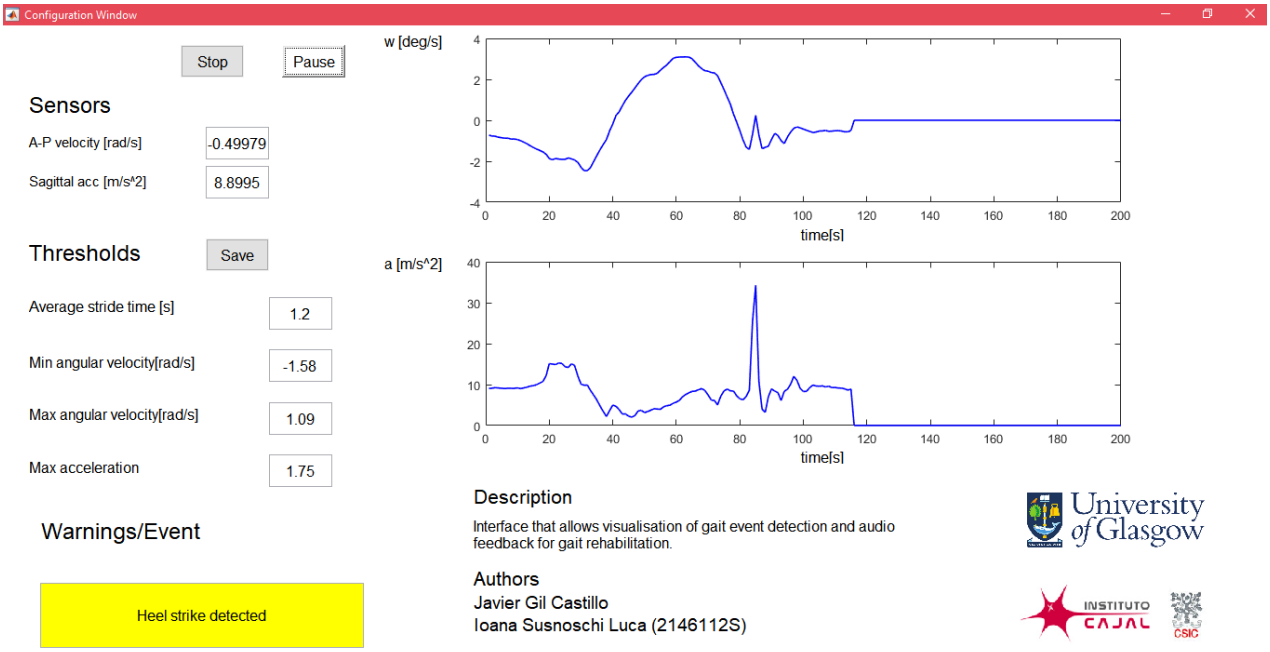
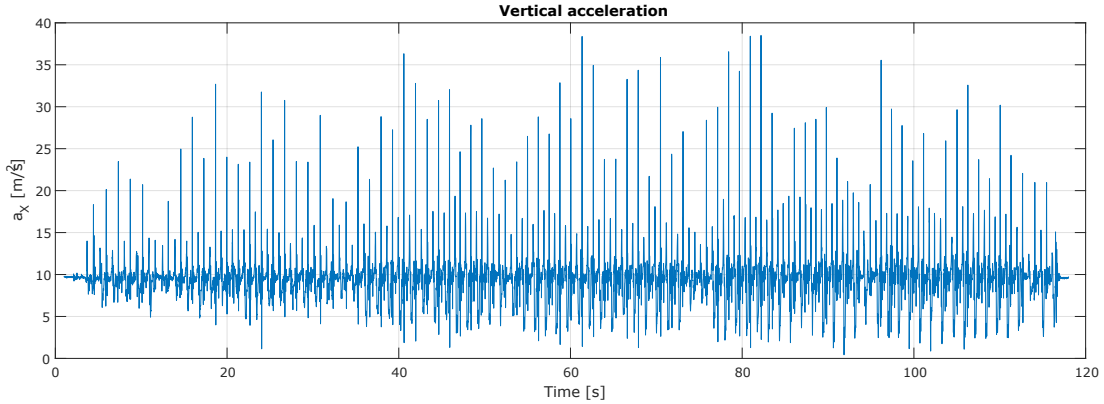
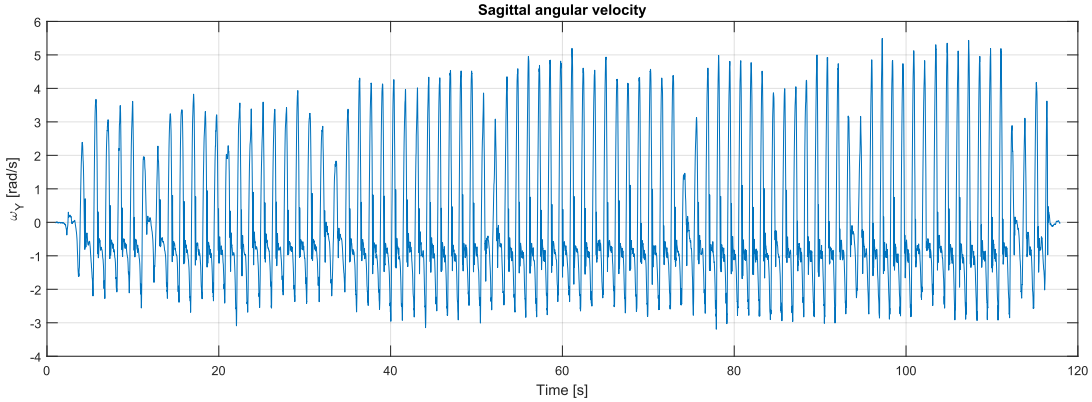


Figure 24: Graphical User Interface displaying the current gait event - heel strike, which corresponds to the first minimum in the angular velocity plot and to the peak in sagittal acceleration plot.

5.2.2 Simulation of real time detection



(a) Vertical acceleration of the subject walking on flat ground.



(b) Sagittal angular velocity

Figure 25: IMU data from one walking trial used in the simulation of real time gait event detection.

For the first part of the study, the relevant physical data acquired from one subject walking on flat ground for two minutes, at their most comfortable speed is presented in Figure 25. The vertical acceleration (Figure 25a) and sagittal angular velocity (Figure 25b) were fed without pre-processing into the S-Function that forms the interface in Figure 24 and contains the detection algorithm.

The local maxima of the sagittal angular velocity were used to find the actual number of strides for each subject, presented in Table 5. The peaks of lower amplitude best identifiable in 25b (e.g. at time=74s) correspond to the subject's change of direction that were not detectable due to the adaptive threshold computation. The algorithm modifications in Section 10.1 accounted for these events and improved the detection accuracy.

The offline detection of IC, TO and MSW, performed in order to cross-check the number of strides found by visual inspection of the plots and to compute the average stride time, as well as the event detection accuracy, is demonstrated in Figure 26. The two middle strides with lower mid-swing amplitude correspond to the change in direction and were correctly identified offline by the detection algorithm. Table 5 contains the values thus obtained for the ten subjects.

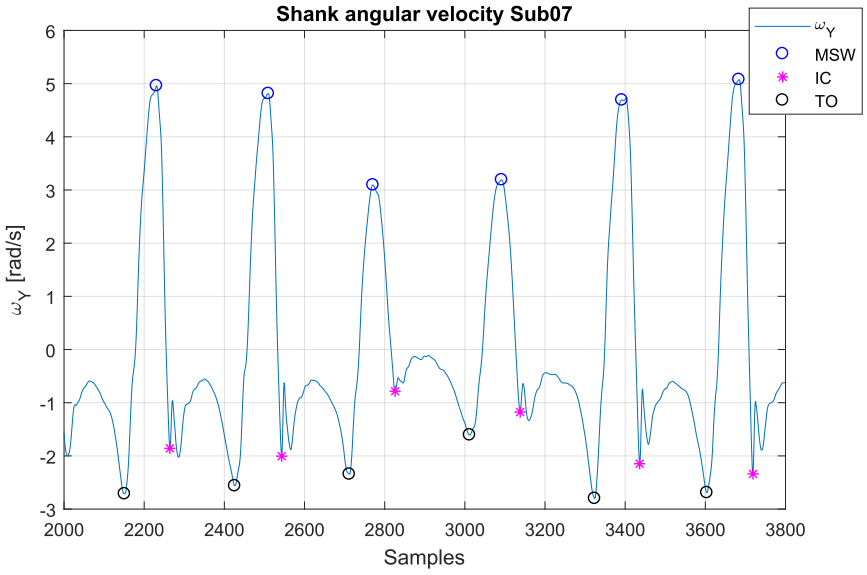


Figure 26: Offline detection of gait events from Gyroscope data used to compute the accuracy of the algorithm, shown in Table 6

Data from the two FSR sensors, analysed as illustrated in Section 4.8.1, confirmed the number of strides found using the offline detection algorithm and presented in Table 5. However, due to the fact that the data was collected at a sampling frequency of $F_s = 10Hz$, the other parameters computed using these signals have low precision and do not provide enough information with respect to the stance and stride times. Parameters calculated from Subject 1 FSR data are presented in Table 4

Table 4: Stride time and stance time of one subject calculated from the two FSR placed on the heel and ball of the foot.

SUBJECT 1		
Average stride time over 20 steps +/- 1SD [s]	Average stance time +/- 1SD [s]	% stance of stride time +/- 1SD
1.4 +/- 0.16	0.8 +/-0.09	60+/-5

Table 5: Actual values from flat ground walking trials.

Subject	Trial time [s]	# strides	Cadence [strides/min]	mean stride time +/-SD [s]
1	118	86	43.68	1.33+/-0.09
2	122	86	42.34	1.38+/-0.08
3	150	120	48.53	1.22+/-0.06
4	177	118	39.88	1.49+/-0.08
5	122	103	50.74	1.26+/-0.08
6	120	97	48.34	1.22+/-0.17
7	131	103	47.34	1.26+/-0.22
8	126	119	56.78	1.03+/-0.04
9	153	118	46.41	1.25+/-0.06
10	148	115	46.77	1.24+/-0.17

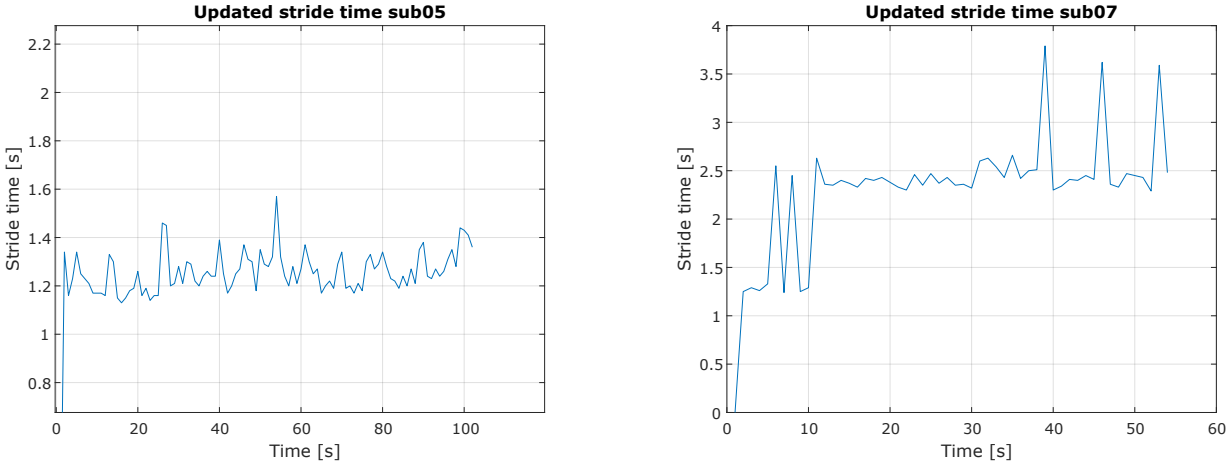
The accuracy of the detection algorithm (Table 6) was obtained by comparing these values to the outputs of the Simulink model. Results from both the initial algorithm and the final version are presented for comparison. However, further Results and the Discussion is based exclusively on the final algorithm version, unless otherwise specified.

Table 6: Accuracy of event detection for individual subjects before and after introducing the vertical acceleration and constraining stride time variation.

2*Subj.	2*Trial time [s]	Detection I			Detection II		
		#strides	% det.	stride_t +/-SD [s]	#strides	% det.	stride_t +/-SD [s]
1	118	79	91.9	1.44+/-0.67	82	95.4	1.37+/-0.33
2	122	74	86.1	1.61+/-0.71	78	90.7	1.49+/-0.48
3	150	64	52.9	2.31+/-0.7	111	92.5	1.3+/-0.37
4	177	102	86.4	1.67+/-0.52	115	97.5	1.52+/-0.31
5	122	102	99.02	1.25+/-0.15	103	100	1.22+/-0.16
6	120	92	94.9	1.23+/-0.43	92	94.9	1.23+/-0.43
7	131	55	53.4	2.32+/-0.63	55	53.4	2.32+/-0.63
8	126	77	64.7	1.54+/-0.54	115	96.6	1.03+/-0.17
9	153	85	72.0	1.70+/-0.68	112	94.9	1.3+/-0.32
10	148	79	68.7	1.78+/-0.69	108	93.9	1.31+/-0.46

Stride time update

The adaptive stride time calculation has yielded value updates similar to the ones plotted in Figure 27. The plots were helpful in identifying the reason for the low detection accuracy of the initial algorithm and for verifying the improvements **brought** about by the additional conditions.



(a) Stride time update for subject 5, whose strides have been detected with 100% accuracy.

(b) Stride time update for subject 7, whose data has been considered an outlier

Figure 27: Adaptive stride time used in the detection algorithm, after improvements. Missed events, especially ICs, have modified the stride time update as shown in Figure 27b.

6 Discussion

The objective of this feasibility study was to test the gait event detection and feedback tool on healthy subjects, with the purpose of characterising the system employed and developing a sound method that can be adapted for the lower limb amputee population. All the aspects included in the study are discussed below, followed by the shortcomings of the tool design and the experimental procedures. A section is then dedicated to suggestions on ways to overcome these and build onto the current version of the gait event detection and biofeedback tool.

6.1 Robotic prosthesis walking trials

As observed in the plot of the collected data from Figure 22, the gait parameters for faulty gait are irregular and have much more variation compared to the normal gait parameters. This was expected as new prosthesis users lack the confidence and experience to control a robotic prosthesis. Heel strike and toe off were identified for each step, following the procedure presented in Section 4.2. However, it can be noticed that the gait event detection for the faulty gait pattern in Figure 22b is correct at the beginning and offset towards the end of the analysed period. This is due to the between-step variation in knee angle and step duration, and emphasises the need for a training tool for new amputees. Additional deviations from target values were also observed when the experienced passive prosthesis user walked normally with the Power Knee. This indicates that training and accommodation period are necessary when switching from a passive or semi-active device to a robotic one.

6.1.1 Load cell data

The ground reaction force curves obtained from the embedded load cell during flat ground walking and plotted in Figure 22 have a different shape from the normal load distribution across gait cycle, presented in literature [99]. The load peaks when the weight is shifted to the prosthetic leg, but decreases gradually as the gait cycle progresses and weight is supported by the ball of the foot. This is most likely due to the use of a passive ankle prosthesis which does not allow suitable plantar flexion and results in an abrupt shifting of the weight to the sound side. The variation in loading the prosthetic knee is visible in the pathological gait condition. Heel strike and toe off points found for each step were used to section the data and obtain an average load across the gait cycle (Figure 23). The curves show overlapped GRF of the seven steps and the computed average. On average, in normal gait 62.2% of the step cycle is stance phase, with a standard deviation of 0.7%, which confirms literature findings and is in line with the 60-40 proportion. For pathological gait, it was found that 53.0% of the gait cycle is stance phase, with a standard deviation of 0.87%. That is, of the total time of 1.42 seconds of normal gait cycle, the stance phase lasts for 0.88 ± 0.01 seconds and the swing- 0.54 seconds. In abnormal gait, the stance phase is 0.76 ± 0.1 seconds of the 1.43 seconds of the gait cycle, with a swing phase of 0.67 seconds. This is equivalent to 5-10% delay in swing initiation for the healthy limb, confirming the hypothesis that new users of lower limb prostheses, specifically Power Knee II, do not perform weight shifting on time. The users could thus benefit from a biofeedback tool for gait training. Variation of step length duration across the 10 seconds was observed in faulty gait. The load applied to the prosthetic knee is up to three times lower compared to values encountered in literature [100].

6.1.2 Knee angular velocity

The disparity between knee velocity in normal gait and pathological gait condition is obvious from the plots in Figure 22. The maximum velocity during normal walk with the prosthesis is 25% lower than the results presented in literature [100], which confirms findings that state that decreased velocity is one of the features of maladaptive gait in lower limb amputee locomotion. Moreover, in the pathological gait condition there is high variation in angular velocity between the steps, and the subject took longer to extend the knee. The data in Table 3 shows the mean and standard deviation of the maximum knee angular velocity during each step. In pathological gait, the average of the maximum knee velocity is less than half of the normal gait velocity, with a very high standard deviation.

6.1.3 Knee angle

In both conditions, the knee angle in extension is higher than zero, which is the healthy gait value. For the normal walking condition, the knee angle in full extension was 8.5°, with a standard deviation of 1.0°, while the pathological gait knee angle in full extension was 8.4°, SD=2.63°. It was reported that the subject maintained the Power Knee slightly flexed in stance phase during walking, however, the cause for this is unknown. The peak flexion of 57.3°, SD=3.8°is slightly lower than the results presented in literature [100, 101]. For pathological gait, the average peak flexion angle is only 29.3°, SD=8.0°. In this case, too, the high standard deviation of the pathological walk condition shows big between-step variation as a characteristic of gait of an untrained user.

6.2 Detection and feedback algorithm

The presented processes aided in developing a tool that detects gait events in real time, with the purpose of providing assistance in the form of audio feedback for gait rehabilitation of prosthesis users. The algorithm development was based on the computational method presented in [84], which employs adaptive threshold calculation to detect six gait events using a single axis gyroscope located on the arch of the foot.

6.2.1 Interface

The sensors' value update in the *Sensors* section was mainly used in the second part of the study, for checking the functioning of the CAN communication. The communication required the data to be unpacked in the Simulink model and passed to the S-Function algorithm. However, due to hardware limitations (personal laptop), as well as Simulink software limitations, the packets of sensor data were not always received or unpacked correctly, meaning that the unpacking order was offset and another parameter was fed to the algorithm. Since the X-axis (vertical) acceleration has a known value, equal to the gravitational acceleration, the Sagittal acceleration field on the interface could be used to verify that the information arriving to the S function was correct at the

beginning of the trials.

The thresholds and stride time, updated after every three strides, indicate the variation between steps. From Table 5 and visual investigation of data it is clear that the stride time variation in healthy gait is minimal. However, in the case of lower limb prosthesis users, the compensatory mechanisms lead to fluctuations in gait parameters such as knee angular velocity and flexion angle (Figure 22b), thereby changing the acceleration and angular velocity of the shank. Moreover, walking with a new prosthesis is often exhausting and may determine the user to increase the strain on the healthy limb and on the muscles of the hip and trunk, which can cause injuries. In these cases, the *Thresholds* section has the purpose of informing the user or rehabilitation assistant on significant changes and act in consequence.

Every time an event was detected, a corresponding announcement was displayed in *Warnings/Event* field, at the bottom section of the interface. Additionally, if the weight was not shifted on time, following the 62-38 gait cycle proportion, a warning was made, which coincided with the audio signal. In the healthy population studied, the event announcements ("Heel Strike", "Toe off", "Mid-swing") were effective in determining the accuracy of the real time gait event detection algorithm. Indicating the three events in a stride might be confusing - it could seem that the Heel Strike ends when Toe Off event occurs, which should be accounted for prior to testing an interface on patients.

For the majority of the subjects, the length of plotted samples (200 samples = 2 seconds) in the *Visualisation of sensor data* section was sufficient to represent at least one and a half steps. Although both the vertical acceleration and the sagittal angular velocity are shown on the interface, the event detection is more straightforward on the latter, as the features of the angular velocity wave form can be distinguished more easily due to the single peak at mid-swing. The peak in X-axis acceleration occurs just before the IC, therefore it aids in identifying the event on the angular velocity plot.

6.2.2 Simulation of real time gait event detection

The detection of gait events on the 963 strides included in the simulation was performed with 95.0% accuracy, meaning that 915 steps out of 963 were detected. Data from subject 7 was excluded as an outlier, having a detection accuracy of only 53.4%m, with an average computed stride time of 2.32±0.63 seconds. Based on the assumptions made for the development of the algorithm, the number of gait events detected in a trial will always be equal, thus the average stride detection accuracy is the same as the detection accuracy of the three individual events (IC, TO, MSW). There have been no false event detections over the ten trials. The initial contact was detected with an average delay of 23±4 ms, the toe off detection occurred with a delay of 20±10ms, and mid-swing events were detected 14±9ms late.

The initial algorithm, which only used the sagittal angular velocity for IC, TO and MSW detection and did not have a stride time variation limit resulted in a much lower detection accuracy - only 79.6% of the strides and events were identified, equivalent to 767 out of 963 strides. Although the gait patterns of the ten healthy subjects did not vary in shape and had small variations in peak amplitude, the way in which each person changed direction differed. Two such patterns were distinguished: 180 degree turn on one heel, with lower swing velocity and lower stride length, and

walking in a U-shape. In both cases, the velocity decreased and the algorithm presented did not always detect turning points as steps. Since the stride time is computed from the time difference between two consecutive ICs, a missed event leads to a doubling of the stride time, which in turn affects further event detections dependent upon known timing in the gait cycle. This was observed from the plots of adaptive stride time computations, such as the ones presented in Figure 27.

To reduce the influence of high stride time variations on detection of further ICs, the distinctive vertical acceleration peak occurring shortly before IC was used as an auxiliary detection mechanism. Additionally, the stride time condition was introduced in order to disregard potential missed event detection when computing the average stride time. The condition assumes that the subject walking at their most comfortable pace will not vary the walking speed by more than 10% in the course of a walking trial, a value which was found after evaluating the offline walking trials and noticing that the duration of a stride varied with 3.8-6.7% across a walking trial. The subjects were not instructed on walking in a straight line or turn and change direction in a certain way in order to preserve as much as possible of real life situation walking patterns.

After including the vertical acceleration and the additional time condition that limited variations in stride time to 10% of the average duration of the ten previous strides, the detection accuracy increased to 95%, a value which is slightly lower compared to similar detection systems presented in literature, of 98-100% [94, 80, 90, 84]. This is, however, compensated for by the low computational cost of the algorithm, which bears significance for the real time implementation. Thus, mounting the algorithm on a microcontroller would allow detection of gait events and feedback in real time, to obtain a fully wearable system. Furthermore, the computational efficiency will be a key factor when including two FSRs to the system in order to increase the detection accuracy, due to the increase in complexity brought about. The accuracy of the detection algorithm is acceptable for the purpose of providing biofeedback for gait training, due to the fact that missed events could, at worst, result in the lack of audio signalling when weight shifting is necessary. However, a similar tool employed for active or semi active prosthesis control would need a 100% detection rate, as the consequences of a missed detection could range from insufficient prosthetic joint flexion to the user falling.

No relationship was found between detection accuracy for individual subjects and their average cadence: the subject whose strides were detected with 100% accuracy had the second highest cadence, 50.7 strides/min, or 101.5 steps/min, while the second highest detection accuracy (97.5%) was obtained from the subject with the lowest cadence, 39.9 strides/min, or 79.8 steps/min. Moreover, the cadence of the subject whose strides were detected only 53.4% of the times and considered an outlier was almost equal to the average cadence of the sample population studied: 47.34 strides/min versus 47.1 strides/min.

The reason for the low detection accuracy from subject 7 walking trial is not known. The only particularity revealed by visual inspection of the data from subject 7 against other subjects, present both in the sagittal angular velocity and in the vertical acceleration wave forms, was a very low amplitude at MSW when changing direction, and a longer time taken to turn by 180°.

The delay in detection of IC, TO and MSW generally ranges between 10-30ms, with less than ten events being registered with up to 90ms latency. None of the gait events have been detected before their actual occurrence. The delays are comparable to the results presented by Maqbool et al. in

[94], where a similar shank IMU setup is used, but up to three times higher than those quoted in Figueiredo's research [84], where the gyroscope is mounted on the foot insole. The average delay in IC detection over the 963 strides was found to be 23±4 ms. As percentage of individual subjects' average stride time, the initial contact was marked up to 1.8% later than the real event. From the point of view of stance time calculation for weight shifting, this could delay the generation of audio feedback. It is still to be determined how significant the delay would be in the context of prosthesis user weight shifting. The toe off was detected with an average delay of 20±10ms, and with a maximum delay of 2.45% of one subject's average stride time duration. Similar to the case of IC detection delay, this may influence the biofeedback timing. Nonetheless, in both cases, a small delay (up to 30ms) in audio signal generation is preferable to a premature feedback, which could be activated unnecessarily before the stance time condition is met.

The mean of the stride time estimations from three consecutive steps, employed in the detection algorithm, was generally overestimated by 0.8-7.9% of the average stride time of each subject, apart from two cases, where the two average values coincided. The overestimation and the higher variation in stride time calculation result from missed IC events, compensated for by using the additional conditions described above. The initial estimation of stride time of 1.2 seconds, chosen empirically, was suitable for the majority of the subjects, although it is an underestimation compared to individual subjects' average stride duration. This can be noticed in Figure 27a, where the value does not change drastically, as well as in the Stride time column of Table 6.

The slight variation in data recording trials (118 to 180s) did not have a visible effect on the quality of the results. The subjects were instructed to walk for two minutes, self paced and with no fixed stopping point, so that they did not feel the need to change their cadence or stride length during the trial.

Force Sensing Resistor data

The initial purpose of including FSR sensors into the gait event detection system was to study their potential to increase the accuracy of stance time calculation for providing biofeedback, as well as enable better event detection. However, the hardware limitations have excluded FSR sensing from the real time testing of the software and have restricted the use for simulated real time event detection. The results presented in Table 4 indicate that a 10Hz sampling frequency does not allow calculation of stance phase percentage of stride time to the resolution required. The algorithm was designed to detect when stance time lasted for longer than 62% of stride time, therefore a much higher sensor resolution would be needed.

Although the real time testing of the IMU-FSR system was not possible as part of the current feasibility study, it was found that data from the two sensors is complementary and can be integrated using a fusion algorithm with the potential to detect more gait events, which would enable identification of gait transitions. Similar systems have been used to detect initial contact, heel off and swing phases [88], using 3 FSRs and a single axis gyroscope, and a more complex setup consisting of 3 IMU sensors and two foot switches was employed for prediction of locomotion intent [81].

The FSR sensors can be used as switches, with boolean output for contact detection. However, the analogue output allows calibration of the contact threshold for individual subjects, thus enabling detection of heel contact and toe contact in subjects with walking defects. The rigidity of the prosthetic foot used by lower limb amputees, which prevents foot ROS (roll over shape), may

result in absence of a clear IC event.

7 Future work

An effective biofeedback tool for gait retraining in the amputee population offers clear, accurate instructions, each time these are needed, and contributes to bettering the user's walking pattern. The gait event detection and audio feedback tool created can be improved in the following aspects:

1. accuracy of the event detection algorithm for the detection of IC, TO, MSW
2. robustness of the detection in real life walking conditions
3. FSR - inertial signal fusion
4. design environment (Simulink)

The algorithm has already been enhanced by using data from vertical acceleration along with sagittal angular velocity, and further investigation into incorporating the other IMU parameters should provide sufficient information for a more complete gait event detection system. To increase the robustness of the detection algorithm, data should be collected in experiments that involve start-stop sequences and instances where the subjects vary their walking speed or mimic asymmetric gait.

Research investigating fusion of complementary sensor data has shown increased accuracy in the context of gait event detection [104], therefore inclusion of force sensors into the real time detection should also be considered. The FSR used in analogue output mode would not only enable computation of stance and stride time, but also allow quick calibration of detection thresholds to account for walking patterns where not all the events are clearly distinguished.

Although testing of the detection and feedback tool under simple walking conditions has yielded satisfactory results, switching to a more suitable real time design environment, such as *C++*, should be considered before a more complex system with increased functionality is developed.

The aim of the designed software was to facilitate learning to walk with a lower limb prosthesis. However, this can be extended to passive ortheses that are used as assistive devices, such as the stance phase knee locking devices.

Use of inertial sensors or fusion of IMU sensors with electrical or mechanical sensors for gait event detection is heavily documented, with promising results for prosthesis control and biofeedback. Nonetheless, as concluded in Section 3, a unified benchmarking system and database would enable more reliable results of the system testing and would reduce the most common limitation encountered in literature: insufficiently large sample size. Therefore, the *Future work* section of the scientific community that develops and tests locomotion instruments should include contribution to such a platform.

8 Conclusions

The hypothesis that increased stance phase on the healthy leg is a pervasive issue faced by the amputee population has been backed by the results of gait pattern analysis from a robotic prosthesis user. The requirements of the biofeedback tool were defined. The system designed to meet them has been developed and assessed as part of a feasibility study that included real time testing and simulating on ten healthy subjects, under two walking conditions.

The accuracy of the gait event detection in simulated real time on the Simulink environment was computed and validated using offline gait analysis and cross-check between IMU and FSR sensor output. The results confirm that the presented tool is able to detect the initial contact, the final contact and the mid swing with satisfactory accuracy, which made it suitable for utilisation in the biofeedback instrument. Although a fusion of FSR and IMU signals for real time gait analysis would offer the potential for increased robustness and expanded capabilities for correction of gait asymmetry, this could not be studied using the presented hardware.
The conclusions of this study indicate that the presented system is a starting point for the construction of a simple and efficient instrument that targets gait asymmetry and could be used outside of rehabilitation centres, as an additional training resource.

9 Project Outcomes

The principal outcomes of this project comprise:

1. Review of literature covering evolution of lower limb prostheses and state of the art in robotic prostheses, to be included in Eurobench2020 documentation.
2. Gait analysis of one Power Knee II user
3. Identification of needs and requirements for a biofeedback tool that would aid in learning to walk with a powered prosthesis
4. Real time communication with the IMU using CAN protocol
5. Design of a gait event detection system that employs a single IMU mounted on the shank, via Matlab Simulink R2017a
6. Investigation of the suitability of an IMU-FES system for gait event detection and audio feedback
7. Testing of the gait event detection and feedback tool on ten healthy subjects walking on flat ground.

References

[1] Online resource. URL: https://www.physio-pedia.com/Gait.

[2] Heydar Sadeghi et al. "Symmetry and limb dominance in able-bodied gait: a review". In: *Gait Posture* 12.1 (2000). DOI: 10.1016/S0966-6362(00)00070-9.

[3] Eric Maupas et al. "Functional asymmetries of the lower limbs. A comparison between clinical assessment of laterality, isokinetic evaluation and electrogoniometric monitoring of knees during walking". In: *Gait posture* 16 (Jan. 2003), pp. 304–12. DOI: 10.1016/S0966-6362(02)00020-6.

[4] Lalit Patnaik. "Dynamics and Control of a Rimless Wheel based 2D Dynamic Walker using Pulsed Torque Actuation". PhD thesis. CERN, 2015.

[5] C. Ross Ethier and Craig A. Simmons. *Introductory Biomechanics - From Cells to Organisms*. Cambridge University Press, 2007. ISBN: 978-0-511-27360-5.

[6] Jaeheung Park. "Synthesis of natural arm swing motion in human bipedal walking". In: *Journal of Biomechanics* 41(7) (2008). DOI: 10.1016/j.jbiomech.2008.02.031.

[7] Liz Tanner. "Biomechanics Lecture 2". Lecture notes University of Glasgow.

[8] *The eight phases of human gait cycle.*

[9] Mena D, Mansour J. M., and Simon S. R. "Analysis and synthesis of human swing leg motion during gait and its clinical applications". In: *Biomechanics* 14 (1981).

[10] Smith J. L. S. and Zernicke R. F. "Predictions for neural control based on limb dynamics". In: *Trend in Neuroscience* 10 (1987).

[11] Winter D. A. and Robertson D. G. E. "Joint torque and energy patterns in normal gait". In: *Biological Cybernetics 29(3):137-42* 29(3) (1978). DOI: 10.1007/BF00337349.

[12] M. P. Kadaba et al. "Repeatability of Kinematic, Kinetic, and Electromyographic Data in Normal Adult Gait". In: *Journal of Orthopaedic Research* 7 (1989).

[13] Ramon Gomez Ferrer-Sapina. "Estudio Biomecanico de la marcha en pacientes co artrosis de cadera". PhD thesis. Universidad de Valencia, Departamento de Medicina, 2005.

[14] Saunders J.B., Inman V.T., and Eberhart H.D. "The major determinants in normal and pathological gait". In: *Journal of Bone and Joint Surgery* 35(3) (1953).

[15] Arthur D. Kuo and J. Maxwell Donelan. "Dynamic Principles of Gait and Their Clinical Implications". In: *Perry Issue: Gait Rehab* 90.2 (2010).

[16] Ismet Handzic and Kyle B. Reed. "Perception of gait patterns that deviate from normal and symmetric biped locomotion". In: *Frontiers in Pshychology* 6.199 (2015). DOI: 10.3389/fpsyg.2015.00199.

[17] D A Winter. "Human balance and posture control during standing and walking". In: *Gait-Posture* 3.4 (1995). DOI: 10.1016/0966-6362(96)82849-9.

[18] Shirota C., Simon A., and Kuiken T. "Transfemoral amputee recovery strategies following trips to their sound and prosthesis sides throughout swing phase." In: *Journal of Neuro-Engineering and Rehabilitation* 12(1) (2015). DOI: 10.1186/s12984-015-0067-8.

[19] Van Velzen J. M. et al. "Physical capacity and walking ability after lower limb amputation: a systematic review". In: *Clinical Rehabilitation* 20(11) (2006). DOI: 10.1177/0269215506070700.

[20] Latanioti E., Angoules A., and Boutsikari E. "Proprioception in Above-the-Knee Amputees with Artificial Limbs". In: *The Scientific World Journal* (2013). DOI: 0.1155/2013/417982.

[21] Katja Orlowski et al. "Analyzing the Transfemoral Amputee Gait using Inertial Sensors-Identifying Gait Parameters for Investigating the Symmetry of Gait - A Pilot Study". In: *Proceedings of the International Conference on Bio-inspired Systems and Signal Processing* (2015). DOI: 10.5220/0005250802580263.

[22] Wong C., Sheppard J., and Williams K. "Improving balance and walking ability in community-dwelling people with lower limb loss: a narrative review with clinical suggestions." In: *Physical Therapy Reviews* 23(2) (2018). DOI: 10.1080/10833196.2018.1451291.

[23] Lee M., Lin C., and Soon K. "Balance control enhancement using sub-sensory stimulation and visual-auditory biofeedback strategies for amputee subjects". In: *Prosthetics and Orthotics International* 31(4) (2007). DOI: 10.1080/03093640601058162.

[24] Winter D. A. and Sienko S. E. "Biomechanics of below-knee amputee gait". In: *Journal of Biomechanics* 21(5) (1988). DOI: 10.1016/0021-9290(88)90142.

[25] J. A. Gilbert et al. "Technical note-auditory feedback of knee angle for amputees". In: *Prosthetics and Orthotics International* 6 (1982).

[26] Michael R TuckerEmail et al. "Control strategies for active lower extremity prosthetics and orthotics: a review". In: *Journal of NeuroEngineering and Rehabilitation* 12:1 (2015). DOI: 10.1186/1743-0003-12-1.

[27] Kannape OA and Herr HM. "Split-belt adaptation and gait symmetry in transtibial amputees walking with a hybrid EMG controlled ankle-foot prosthesis". In: *Conf Proc IEEE Eng Med Biol Soc.* (2016). DOI: 10.1109/EMBC.2016.7591964..

[28] Mazuquin B. F. et al. "Kinematic Gait Analysis Using Inertial Sensors with Subjects after Stroke in Two Different Arteries". In: *Journal of Physical Therapy Science* 26(8) (2014). DOI: 10.1589/jpts.26.1307.

[29] Zambarbieri D., Schmid M., and Verni G. "Sensory feedback for lower limb prostheses". In: *Intelligent Systems and Technologies in Rehabilitation Engineering* (2000). DOI: 10.1201/9781420042122.

[30] Goldfarb M., Lawson B., and Shultz A. "Realizing the Promise of Robotic Leg Prostheses". In: *Science Translational Medicine* 5(210) (2013). DOI: 10.1126/scitranslmed.3007312.

[31] Chen B., Zheng E., and Wang Q. "A locomotion intent prediction system based on multi-sensor fusion". In: *Sensors (Basel)* 14(7) (2014). DOI: 10.3390/s140712349..

[32] Yi-Chung Lin, Margit Gföhler, and Marcus G. Pandy. "Quantitative Evaluation Of The Major Determinants Of Human Gait". In: *Journal of Biomechanics* 47 (Apr. 2014). DOI: 10.1016/j.jbiomech.2014.02.002.

[33] Kaufman K., Frittoli S., and Frigo C. "Gait asymmetry of transfemoral amputees using mechanical and microprocessor-controlled prosthetic knees". In: *Clinical Biomechanics* 27(5) (2012). DOI: 10.1016/j.clinbiomech.2011.11.011.

[34] Rine R. M. et al. "Vestibular function assessment using the NIH Toolbox". In: *Neurology* 80 (2013). DOI: 10.1212/wnl.0b013e3182872c6a.

[35] Maetzler W. et al. "Impaired Trunk Stability in Individuals at High Risk for Parkinson's Disease". In: *PLoS ONE* 7(3) (2012). DOI: 10.1371/journal.pone.0032240.

[36] Mancini M. et al. "ISway: a sensitive, valid and reliable measure of postural control". In: *Journal of NeuroEngineering and Rehabilitation* 9(1) (2012). DOI: 10.1186/1743-0003-9-59.

[37] Tong K. and Granat M. H. "A practical gait analysis system using gyroscopes". In: *Medical Engineering Physics* 21(2) (1999). DOI: 10.1016/s1350-4533(99)00030-2.

[38] Maetzler W. et al. "Impaired Trunk Stability in Individuals at High Risk for Parkinson's Disease". In: *PLoS ONE* 7(3) (2012). DOI: 10.1371/journal.pone.0032240.

[39] Henry Chambers and David H Sutherland. "A Practical Guide to Gait Analysis". In: *The Journal of the American Academy of Orthopaedic Surgeons* 10 (May 2002), pp. 222–31. DOI: 10.5435/00124635-200205000-00009.

[40] O'Reilly M. et al. "Wearable Inertial Sensor Systems for Lower Limb Exercise Detection and Evaluation: A Systematic Review". In: *Sports Medicine* 48(5) (2018). DOI: 10.1007/s40279-018-0878-4.

[41] Leardini A. et al. "Validation of the angular measurements of a new inertial-measurement-unit based rehabilitation system: comparison with state-of-the-art gait analysis". In: *Journal of NeuroEngineering and Rehabilitation* 11(1) (2014). DOI: 10.1186/1743-0003-11-136.

[42] Giggins O. M., Sweeney K. T., and Caulfield B. "Rehabilitation exercise assessment using inertial sensors: a cross-sectional analytical study". In: *Journal of NeuroEngineering and Rehabilitation* 11(1) (2014). DOI: 10.1186/1743-0003-11-158.

[43] Tang Z. et al. "Measurement and Estimation of 3D Orientation using Magnetic and Inertial Sensors". In: *Advanced Biomedical Engineering* 4(0) (2015). DOI: 10.14326/abe.4.135.

[44] Lin J. F. S. and Kulić D. "Human pose recovery using wireless inertial measurement units". In: *Physiological Measurement* 33(12) (2012). DOI: 10.1088/0967-3334/33/12/2099.

[45] Coley B. et al. "Stair climbing detection during daily physical activity using a miniature gyroscope". In: *Gait Posture* 22(4) (2005). DOI: 10.1016/j.gaitpost.2004.08.008.

[46] Tong K. and Granat M. H. "A practical gait analysis system using gyroscopes". In: *Medical Engineering Physics* 21(2) (1999). DOI: 10.1016/s1350-4533(99)00030-2.

[47] He Huang et al. "Continuous Locomotion-Mode Identification for Prosthetic Legs Based on Neuromuscular–Mechanical Fusion". In: *Transactions on Biomedical Engineering* 58(10) (2010). DOI: 10.1109/tbme.2011.2161671.

[48] José-Antonio Gil-Gómez et al. "Effectiveness of a Wii balance board-based system (eBaViR) for balance rehabilitation: a pilot randomized clinical trial in patients with acquired brain injury". In: *Journal of NeuroEngineering and Rehabilitation* 8 (2011).

[49] Marcos Duarte and Sandra M. S. F. Freitas. "Revision of posturography based on force plate for balance evaluation". In: *Revista Brasileira de Fisioterapia* 3 (2010). ISSN: 1413-3555.

[50] Can Tunca et al. "Inertial Sensor-Based Robust Gait Analysis in Non-Hospital Settings for Neurological Disorders". In: *Sensors* 17(825) (2017). DOI: 10.3390/s17040825.

[51] "Development of a low cost force platform for biomechanical parameters analysis". In: *Research on Biomedical Engineering* 3 (2017). DOI: 0.1590/2446-4740.01217.

[52] Shaikh Muhammad Faraz, Zoran Salcic, and Kevin Wang. "Analysis and Selection of the Force Sensitive Resistors for Gait Characterisation". In: *6th International Conference on Automation, Robotics and Applications*. 2015.

[53] Richard Baker. "Gait analysis methods in rehabilitation". In: *Journal of NeuroEngineering and Rehabilitation* 3 (2006). DOI: 10.1186/1743-0003-3-4.

[54] Michael R. Tucker, Jeremy Olivier, and Anna Pagel et al. "Control strategies for active lower extremity prosthetics and orthotics: a review". In: *Journal of NeuroEngineering and Rehabilitation* 12 (2015).

[55] Edgar Charry et al. “A study on band-pass filtering for calculating foot displacements from accelerometer and gyroscope sensors”. In: *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference* 2009 (Jan. 2009), pp. 4824, 4826–7. DOI: 10.1109/IEMBS.2009.5332673.

[56] Yoong Ping Lim, Yi-Chung Lin, and Marcus G. Pandy. “Effects of step length and step frequency on lower-limb muscle function in human gait”. In: *Journal of Biomechanics* 57 (Mar. 2017). DOI: 10.1016/j.jbiomech.2017.03.004.

[57] Tom Kepple, K Lohmann Sicgel, and Steven Stanhope. “Relative contributions of the lower extremity joint moments to forward progression and support during gait”. In: *Gait Posture - GAIT POSTURE* 4 (Apr. 1996), pp. 178–178. DOI: 10.1016/0966-6362(96)80596-0.

[58] Colin L. Eakin, Peter Quesada, and Harry Skinner. “Lower-Limb Proprioception in Above-Knee Amputees”. In: *Clinical orthopaedics and related research* NA; (Dec. 1992), pp. 239–46. DOI: 10.1097/00003086-199211000-00034.

[59] Richard E. Seroussi et al. “Mechanical work adaptions of above-knee amputee ambulation”. In: *Archives of physical medicine and rehabilitation* 77 (Dec. 1996), pp. 1209–14. DOI: 10.1016/S0003-9993(96)90151-3.

[60] M. A. B. Husman et al. “A Wearable Skin Stretch Haptic Feedback Device: Towards Improving Balance Control in Lower Limb Amputees”. In: *Conf Proc IEEE Eng Med Biol Soc* (2016). DOI: 10.1109/EMBC.2016.7591147.

[61] Chen B, Feng Y, and Wang Q. “Combining Vibrotactile Feedback with Volitional Myoelectric Control for Robotic Transtibial Prostheses”. In: *Frontiers in Neurorobotics* 10 (2016). DOI: 10.3389/fnbot.2016.00008.

[62] Sigrist R. et al. “Augmented visual, auditory, haptic, and multimodal feedback in motor learning: A review”. In: *Psychonomic Bulletin Review* 20(1) (2012). DOI: 10.3758/s13423-012-0333-8.

[63] Matjaĉić Z. and Burger H. “Dynamic balance training during standing in people with transtibial amputation: A pilot study.” In: *Prosthetics and Orthotics International* 17(3) (2013). DOI: 10.1080/03093640308726684.

[64] Online. From Wikimedia Commons, the free media repository. URL: https://commons.wikimedia.org/wiki/File:Anatomical_Planes-en.svg.

[65] Kavounoudias A. et al. “Bilateral changes in somatosensory sensibility after unilateral below-knee amputation.” In: *Archives of Physical Medicine and Rehabilitation* 86(4) (2005). DOI: 10.1016/j.apmr.2004.10.030.

[66] Buckley J. et al. "Understanding adaptive gait in lower-limb amputees: insights from multivariate analyses". In: *Journal of NeuroEngineering and Rehabilitation* 10(1) (2013). DOI: 10.1186/1743-0003-10-98..

[67] Sonja M.H.J. Jaegers, Hans Arendzen, and Henry J. de Jongh. "Prosthetic gait of unilateral transfemoral amputees: A kinematic study". In: *Archives of physical medicine and rehabilitation* 76 (Sept. 1995), pp. 736–43. DOI: 10.1016/S0003-9993(95)80528-1.

[68] D.A. Winter and H.J. Yack. "EMG profiles during normal human walking: stride-to-stride and inter-subject variability". In: *Electroencephalography and clinical Neurophysiology* 67 (1987).

[69] *Recommended setup for a 16-IMUs network on the human body*. Online. URL: www.technaid.com.

[70] Online. URL: https://www.ossur.co.uk/prosthetic-solutions/products/dynamic-solutions/power-knee.

[71] Peter Gabriel Adamczyk and Arthur D Kuo. "Mechanisms of Gait Asymmetry Due to Push-off Deficiency in Unilateral Amputees". In: (). DOI: 10.1109/TNSRE.

[72] Javier Gil Castillo. MA thesis.

[73] Introduction to CAN. URL: http://www.ni.com/white-paper/2732/.

[74] Technaid. *Inertial Measurement Unit Tech imu V.4 - Standard CAN*.

[75] *Windows Software for Displaying CAN and CAN FD Messages*. Online. URL: https://www.peak-system.com/PCAN-View.242.0.html?&L=1.

[76] *Bug report Matlab 2017a*. URL: https://uk.mathworks.com/support/bugreports/1533668?s_tid=srchtitle.

[77] Muhammad Afif Husman et al. *Portable Haptic Device for Lower Limb Amputee Gait Feedback: Assessing Static and Dynamic Perceptibility*. 2017. ISBN: 9781538622964. DOI: 10.0/Linux-x86_64.

[78] Mohammed A. Azhar, Darwin Gouwanda, and Alpha A. Gopalai. "Development of an intelligent real-time heuristic-based algorithm to identify human gait events". In: *2014 IEEE-EMBS International Conference on Biomedical and Health Informatics, BHI 2014*. 2014. ISBN: 9781479921317. DOI: 10.1109/BHI.2014.6864429.

[79] Gianluca Bonora et al. "A new instrumented method for the evaluation of gait initiation and step climbing based on inertial sensors: A pilot application in Parkinson's disease". In: *Journal of NeuroEngineering and Rehabilitation* (2015). ISSN: 17430003. DOI: 10.1186/s12984-015-0038-0.

[80] Paola Catalfamo, Salim Ghoussayni, and David Ewins. "Gait event detection on level ground and incline walking using a rate gyroscope". In: *Sensors* (2010). ISSN: 14248220. DOI: 10.3390/s100605683.

[81] Baojun Chen, Enhao Zheng, and Qining Wang. "A locomotion intent prediction system based on multi-sensor fusion". In: *Sensors (Switzerland)* (2014). ISSN: 14248220. DOI: 10.3390/s140712349.

[82] Christopher Ferrigno et al. "The Feasibility of Using Augmented Auditory Feedback From a Pressure Detecting Insole to Reduce the Knee Adduction Moment: A Proof of Concept Study". In: *Journal of Biomechanical Engineering* (2016). ISSN: 0148-0731. DOI: 10.1115/1.4032123.

[83] Inc 1994-2019 The MathWorks. URL: https://uk.mathworks.com/help/simulink/sfg/what-is-an-s-function.html.

[84] Joana Figueiredo et al. "Gait Event Detection in Controlled and Real-Life Situations: Repeated Measures From Healthy Subjects". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* PP (Aug. 2018), pp. 1–1. DOI: 10.1109/TNSRE.2018.2868094.

[85] Blair Hu, Elliott Rouse, and Levi Hargrove. "Fusion of Bilateral Lower-Limb Neuromechanical Signals Improves Prediction of Locomotor Activities". In: *Frontiers in Robotics and AI* (2018). ISSN: 2296-9144. DOI: 10.3389/frobt.2018.00078.

[86] He Huang et al. "Continuous locomotion-mode identification for prosthetic legs based on neuromuscular - Mechanical fusion". In: *IEEE Transactions on Biomedical Engineering* (2011). ISSN: 00189294. DOI: 10.1109/TBME.2011.2161671. arXiv: NIHMS150003.

[87] Deepak Joshi and Michael E. Hahn. "Terrain and Direction Classification of Locomotion Transitions Using Neuromuscular and Mechanical Input". In: *Annals of Biomedical Engineering* (2016). ISSN: 15739686. DOI: 10.1007/s10439-015-1407-3.

[88] Peter Shull et al. "Toe-in gait reduces the first peak knee adduction moment in patients with medial compartment knee osteoarthritis". In: *Journal of biomechanics* 46 (Nov. 2012). DOI: 10.1016/j.jbiomech.2012.10.019.

[89] Justin J. Kavanagh. "Lower trunk motion and speed-dependence during walking". In: *Journal of NeuroEngineering and Rehabilitation* (2009). ISSN: 17430003. DOI: 10.1186/1743-0003-6-9.

[90] Elissa Ledoux. *Inertial Sensing for Gait Event Detection and Transfemoral Prosthesis Control Strategy*. 2018. DOI: 10.1109/TBME.2018.2813999.

[91] Hyo Ki Lee et al. "Novel algorithm for the hemiplegic gait evaluation using a single 3-axis accelerometer". In: *Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Engineering the Future of Biomedicine, EMBC 2009*. 2009. ISBN: 9781424432967. DOI: 10.1109/IEMBS.2009.5333650.

[92] *Recommended setup for a 16-IMUs network on the human body*. Online. URL: http://www.technaid.com/wp-content/uploads/2016/05/Recommended-setup-for-a-16-IMUs-network-on-the-human-body-EN.pdf.

[93] Avril Mansfield and Gerard M. Lyons. "The use of accelerometry to detect heel contact events for use as a sensor in FES assisted walking". In: *Medical Engineering and Physics* (2003). ISSN: 13504533. DOI: 10.1016/S1350-4533(03)00116-4.

[94] H. F. Maqbool et al. "A Real-Time Gait Event Detection for Lower Limb Prosthesis Control and Evaluation". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* (2017). ISSN: 15344320. DOI: 10.1109/TNSRE.2016.2636367.

[95] Farhan Maqbool et al. "Real-time gait event detection for lower limb amputees using a single wearable sensor". In: vol. 2016. Aug. 2016, pp. 5067–5070. DOI: 10.1109/EMBC.2016.7591866.

[96] Yerzhan Massalin, Madina Abdrakhmanova, and Huseyin Atakan Varol. "User-independent intent recognition for lower limb prostheses using depth sensing". In: *IEEE Transactions on Biomedical Engineering* (2018). ISSN: 15582531. DOI: 10.1109/TBME.2017.2776157.

[97] Kyung Ryoul Mun et al. "Gait Estimation from Anatomical Foot Parameters Measured by a Foot Feature Measurement System using a Deep Neural Network Model". In: *Scientific Reports* (2018). ISSN: 20452322. DOI: 10.1038/s41598-018-28222-2.

[98] Michelle Norris, Ian C Kenny, and Ross Anderson. *COMPARISON OF ACCELEROMETRY STRIDE TIME CALCULATION METHODS*. Tech. rep.

[99] Berceanu C. et al. "Gait analysis parameters of healthy human subjects with asymmetric loads". In: *Computer Methods in Biomechanics and Biomedical Engineering* 19(8) (2015). DOI: 10.1080/10255842.2015.1075008.

[100] Aaron J Young et al. "Intent Recognition in a Powered Lower Limb Prosthesis Using Time History Information". In: *Annals of Biomedical Engineering* 42.3 (2014). DOI: 10.1007/s10439-013-0909-0.

[101] Ann M Simon et al. "Configuring a Powered Knee and Ankle Prosthesis for Transfemoral Amputees within Five Specific Ambulation Modes". In: *PLOS One* 9.6 (2014).

[102] Domen Novak and Robert Riener. "A survey of sensor fusion methods in wearable robotics". In: *Robotics and Autonomous Systems*. 2015. ISBN: 0921-8890. DOI: 10.1016/j.robot.2014.08.012.

[103] Kunihiro Ogata et al. “Gait training assist system of a lower limb prosthetic visualizing muscle activation pattern using a color-depth sensor”. In: *IEEE International Conference on Rehabilitation Robotics*. 2017. ISBN: 9781538622964. DOI: 10.1109/ICORR.2017.8009249.

[104] Khaleghi B. et al. “Multisensor data fusion: A review of the state-of-the-art”. In: *Information Fusion* 14(1) (2013). DOI: 10.1016/j.inffus.2011.08.001.

article
listings color [margin=2cm]geometry

10 Appendix

10.1 S function for gait event detection and graphical user interface

```
function [sys,x0,str,ts,simStateCompliance] = essential_SFun(t,x,u,
    flag)


global SjWin
global contTime
global fs


% Data entry variables that correspond to IMU sensor values
    global dataAccX dataAccY dataGyroX dataGyroY

% variables related to gait events - checks
    global acc_max_x
    global IC_current_index IC_current MSW_current MSWindex MSW_switch
        TO_current TO_current_index
    global sample_count
    % counter to recalculate the thresholds and stride time every 3
       strides
    global step_count_3
    % Thresholds
    global dataThrAccMax dataThrAccMin dataThrGyroMax dataThrGyroMin
       dataStride
        % Derivatives of variables
        global gyro_prev gyro_2prev diffvel_y diffvel_y_pre diffacc_x
           diffacc_x_pre

    % Counters & buffers    [ not all shown]
    global stride_t stride_t_buffer total_step_counter
       first_step_detected
    global first_event_counter
    global IC_index_buffer IC_value_buffer TO_index_buffer
       TO_value_buffer MSW_index_buffer MSW_value_buffer

        % threshold values for edit field update
        global thrGyroMax thrGyroMin thrAccMax
        % Warnings on interface
        global dataWarning
```

```
% Output variables
    global numOutputs
    global dataOut
    global feedback

switch flag,

    %%%%%%%%%%%%%%%%%%
    % Initialisation %
    %%%%%%%%%%%%%%%%%%
    case 0,
        [sys,x0,str,ts,simStateCompliance]=mdlInitializeSizes;

        %% Variables S-func.
        tic
        fs          = 64;
        contTime     = 0;

        %% Define figure
        WindowPosition          = [0.05 .05 .8 .8];
        SjWin.figHandle           = figure('Units', 'normalized', ...
            'position', WindowPosition, ...
            'menubar', 'none', ...
            'Name', 'Configuration Window', ...
            'color', 'w',...
            'NumberTitle','off',...
            'Toolbar','none');

        %% Define Axes and points

        % Graphic 1
        SjWin.p1Axes = axes('Parent',SjWin.figHandle,...
            'xlim',[0 200],'ylim',[0 200],'Color','w',...
            'drawmode','fast','Units','Normalized','Position',[0.38
              0.73 0.50 0.25]);

        plotIndex = 1;

        buffy1 = zeros(2,200);

        SjWin.plot1 = plot(1:1:200,buffy1(1,:),'k','LineWidth',1.2);

        % Graphic 2
        SjWin.p2Axes = axes('Parent',SjWin.figHandle,...
            'xlim',[0 200],'ylim',[0 200],'Color','w',...
            'drawmode','fast','Units','Normalized','Position',[0.38
              0.39 0.50 0.25]);
```

```
77
78          buffy2 = zeros(2,200);
79          SjWin.plot2 = plot(1:1:200,buffy2(1,:),'k','LineWidth',1.2);
80
81
82  %% Images
83
84          % Creation of axes for images
85          SjWin.faceAxes1 = axes('Parent',SjWin.figHandle, 'Color','w'
                ,...
86              'drawmode','fast','Position',[0.8 0.03 0.15 0.15]);
87          SjWin.faceAxes2 = axes('Parent',SjWin.figHandle, 'Color','w'
                ,...
88              'drawmode','fast','Position',[0.8 0.17 0.15 0.15]);
89
90  %
91          % Load images from computer
92          SjWin.Paradigm.csic = imread('C:\Users\ioana\Desktop\Uni\MEng
                Project\IMU\simulateRT\csic.jpg');
93          SjWin.Paradigm.glasgow = imread('C:\Users\ioana\Desktop\Uni\
                MEng Project\IMU\simulateRT\glasgow.jpg');
94
95          % Show images in axes creted anteriorly
96          axes(SjWin.faceAxes1);
97          SjWin.Paradigm.imag1 = imshow(SjWin.Paradigm.csic);
98          axes(SjWin.faceAxes2);
99          SjWin.Paradigm.imag2 = imshow(SjWin.Paradigm.glasgow);
100
101
102         %% Define PushButtons
103
104         % [ not in essential form ]
105
106
107         SjWin.buttonStop = uicontrol('Style', 'pushbutton',...
108             'String', 'Stop','Units','Normalized','Position', [0.14
                  0.92 0.05 0.05],'FontSize',12,...
109             'Callback', @callBack_PushButtonStop);
110
111         SjWin.buttonSave = uicontrol('Style', 'pushbutton',...
112             'String', 'Save','Units','Normalized','Position', [0.16
                  0.625 0.05 0.05],'FontSize',12,...
113             'Callback', @callBack_PushButtonSave);
114
115         SjWin.buttonStart = uicontrol('Style', 'pushbutton',...
116             'String', 'Start','Units','Normalized','Position', [0.22
                  0.92 0.05 0.05],'FontSize',12,...
```

```
117             'Callback', @callBack_PushButtonStart);
118
119         %% Define edits
120
121         % Initialise variables corresponding to edits
122         % Sensores
123         dataAccX = 0;
124         dataAccY = 0;
125         dataGyroX = 0;
126         dataGyroY = 0;
127         % Initial thresholds found heuristically - for display on
                interface
128         dataStride = 1.2;           % average stride time found in
                findPeaks
129         dataThrGyroMin = -1.58;     % rad/s
130         dataThrGyroMax = 1.09;      % rad/s
131         dataThrAccMax  = 1.75;        % m/s^2
132         % Thresholds for detection algorithm
133         thrGyroMax = 1.09;
134         thrGyroMin = -1.58;
135         thrAccMax = 1.75;
136         stride_t = 1.2;
137       % Buffers for acceleration and gyroscope data
138         buffSize = 3;
139         acc_x_buffer = zeros(1, 3);          % same for gyro y buffer
140         stride_t_buffer = zeros(1,200);
141         IC_index_buffer = zeros(1,500);      % same for MSW, TO
142         IC_value_buffer = zeros(1,500);
143
144         % Warnings
145         dataWarning = 'Ready';
146
147         % Outputs
148         numOutputs = 7;          % IC, TO, MSW (val+index); output = 1
                for audio feedback and 0 for nothing
149         dataOut = zeros(1,numOutputs);
150
151         % Edit fields    [ not all shown]
152         SjWin.editWarnings = uicontrol('Style', 'edit',...
153             'String', num2str(dataWarning),'Units','Normalized','
                    Position', [0.03 0.05 0.255 0.1],'FontSize',12,...
154             'BackgroundColor','white','Callback',
                    @callBack_EditWarnings);
155
156         SjWin.editGyroY = uicontrol('Style', 'edit',...
157         'String', num2str(dataGyroY),'Units','Normalized','Position',
                [0.16 0.795 0.05 0.05],'FontSize',12,...
```

```
        'BackgroundColor','white','Callback', @callBack_EditGyroY);


        %% Define static texts

        % Informative display - names, axis labels   [ not all shown]
        SjWin.txtSensors = uicontrol('Style','text',...
            'Units','Normalized','Position',[0.02 0.86 0.1 0.04],'HorizontalAlignment','left','FontSize',18,...
            'String','Sensors','BackgroundColor','white','ForegroundColor','black');

        SjWin.txtAuthorName2 = uicontrol('Style','text',...
            'Units','Normalized','Position',[0.37 0.003 0.25 0.1],'HorizontalAlignment','left','FontSize',14,...
            'String','Ioana Susnoschi Luca (2146112S)','BackgroundColor','white','ForegroundColor','black');

        %% Intro
        %%%%%%%%%%%
        % Output  %
        %%%%%%%%%%%
    case 3,
        % %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        sys=mdlUpdate(t,x,u);

        %--- contTime is always either increasing or reseted -- %
        contTime          = contTime+1;

        %% Edit values

        % Obtaining and storing data corresponding to each sensor
        % from S function inputs
        pause(0.01)
        dat1 = u(1);            % acceleration x-axis for display DO NOT normalise wrt g
        dat2 = u(2);            % acceleration y-axis
        dat3 = u(3);            % gyro x-axis
        dat4 = u(4);            % gyro y-axis

        acc_x_buffer = [u(1)/9.81, acc_x_buffer(1:2)];
        acc_y_buffer = [u(2), acc_y_buffer(1:2)];
        gyro_x_buffer = [u(3), gyro_x_buffer(1:2)];
        gyro_y_buffer = [u(4), gyro_y_buffer(1:2)];

```

```
198         % Update interface sensor and threshold data everytime the
                value changes
199         if(dat1 ~= dataAccX)
200             dataAccX = dat1;
201             set(SjWin.editAccX,'String',num2str(dataAccX))
202         end
203
204         if(dat4 ~= dataGyroY)
205             dataGyroY = dat4;
206             sample_count = sample_count + 1;
207             set(SjWin.editGyroY,'String',num2str(dataGyroY))
208         end
209
210         if(thrGyroMax ~= dataThrGyroMax)
211             dataThrGyroMax = thrGyroMax;
212             set(SjWin.editThrGyroMax,'String',num2str(dataThrGyroMax))
213         end
214
215         if(thrGyroMin ~= dataThrGyroMin)
216             dataThrGyroMin = thrGyroMin;
217             set(SjWin.editThrGyroMin,'String',num2str(dataThrGyroMin))
218         end
219
220         if(thrAccMax ~= dataThrAccMax)
221             dataThrAccMax = thrAccMax;
222             set(SjWin.editThrAccMax,'String',num2str(dataThrAccMax))
223         end
224
225         if(stride_t ~= dataStride)
226             dataStride = stride_t;
227             set(SjWin.editStride,'String',num2str(dataStride))
228         end
229
230             % Configure interface when START button is pressed
231             % START button on
232             if(start == 1)
233
234                 % Pressing the START button changes the buttons's
                        colour to
235                 % indicate its state and initialise plotIndex variable
                        to 0
236                 % to start plotting of data from sensors, from the
                        origin.
237
238                 if (controlStart == 0)
239                     % Change value to prevent if loop from running
                            again
```

```
240                     controlStart = 1;
241                     plotIndex = 1;
242                     set(SjWin.buttonStart,'BackgroundColor','white','
                            String','Pause','ForegroundColor','black')
243
244                 end
245
246                 % Calculate current and previous derivatives of AccX
                        and Gyro
247                 check_max_fct = 0;
248                 diffvel_y = diffVelocity(gyro_y_buffer(1,1),
                        gyro_y_buffer(1,2));
249                 diffvel_y_pre = diffVelocity(gyro_y_buffer(1,2),
                        gyro_y_buffer(1,3));
250                 diffacc_x = diffVelocity(acc_x_buffer(1,1),
                        acc_x_buffer(1,2));
251                 diffacc_x_pre = diffVelocity(acc_x_buffer(1,2),
                        acc_x_buffer(1,3));
252
253                 % HAVE TO DETECT angular velocity MAXIMUM (MSW) ONCE
                        before
254                 % event detection starts, to signal that subject is
                        walking
255                 if switch_first == 0
256                     first_step_detected = FirstMaxDet(diffvel_y,
                            diffvel_y_pre, gyro_y_buffer(1,2));
257                 end
258                 acc_max_x = MaxAccFct(diffacc_x, diffacc_x_pre,
                        acc_x_buffer(1,2));
259
260
261                 if first_step_detected == 1
262 %                     set(SjWin.editWarnings,'BackgroundColor','yellow
    ','String','it is starting')
263                     switch_first = 1;
264                     time_to_IC = sample_count;
265
266 %                     first_step_detected = 1;
267                     %max_counter = 0;          % so it only enters this
                            if once
268
269                     % check for intial contact
270                     if (diffvel_y > 0 & diffvel_y_pre < 0 &
                            gyro_y_buffer(1,2)<0)
271                         % check for false detections due to jitter
272                         if ((sample_count - IC_current_index > 0.5*
                                stride_t*100) | (abs(sample_count -
```

```
                                buffer_acc_max_index(1,1)) < 10 &
                                TO_current_index - acc_max_index> 0.4*
                                stride_t)) & IC_switch == 0
273
274                              IC_current = gyro_y_buffer(1,2);
275                              IC_current_index = sample_count+1;
276                              IC_index_buffer = [IC_current_index,
                                    IC_index_buffer(1:499)];
277                              IC_value_buffer = [IC_current,
                                    IC_value_buffer(1:499)];
278                              IC_switch = 1;
279                              if IC_index_buffer(1,1) ~= IC_index_buffer
                                    (1,2)
280                                  stride_t_buffer = [(IC_index_buffer
                                        (1,1) - IC_index_buffer(1,2))/100,
                                        stride_t_buffer(1:199)];
281                              end
282
283                              % ADAPTIVE threshold and stride time
284                              % calculation
285                              if step_count_3 == 3
286                                  stride_t = Stride_timeUpdate(
                                        IC_index_buffer(1,1),IC_index_buffer
                                        (1,4));
287
288                                  % ADDITIONAL CONDITION FOR FREE
                                        WALKING.
289 %                                    if stride_t > 1.15*mean(
    stride_t_buffer(2:11)) & total_step_counter > 12
290 %                                        stride_t = mean(stride_t_buffer
    (2:11));
291 %                                    end
292 %                                    stride_t_buffer = [stride_t,
    stride_t_buffer(1:199)];
293 %                                    if stride_t > mean(stride_t_buffer
    (1:5)) + 3*std(stride_t_buffer(1:5))
294 %                                        stride_t = 1.4;     % restore to
     default
295 %                                    end
296
297                                  thrGyroMax = ThresholdUpdate(
                                        buffer_gyro_max_value(1,1),
                                        buffer_gyro_max_value(1,2),
                                        buffer_gyro_max_value(1,3), 0.6);
298                                  thrGyroMin = ThresholdUpdate(
                                        buffer_gyro_min_value(1,1),
                                        buffer_gyro_min_value(1,2),
```

```
                    buffer_gyro_min_value(1,3), 0.6);
299                 thrAccMax  = ThresholdUpdate(
                    buffer_acc_max_value(1,1),
                    buffer_acc_max_value(1,2),
                    buffer_acc_max_value(1,3), 0.75);
300                 % reset counter
301                 step_count_3 = 0;
302             end
303
304             step_count_3 = step_count_3 + 1;
305             total_step_counter = total_step_counter +
                    1;
306         end
307
308     end
309
310     if IC_current < 0
311         set(SjWin.editWarnings,'BackgroundColor','
                yellow','String','Heel strike detected')
312
313         %search for TO event as minimum in sagittal
                angular
314         %velocity
315         TO_current = MinGyroFct(diffvel_y,
                diffvel_y_pre, gyro_y_buffer(1,2));
316         if sample_count - IC_current_index > 0.62*
                stride_t*100 & TO_current > 0
317             feedback = 1;
318             set(SjWin.editWarnings,'BackgroundColor','
                    yellow','String','Shift weight!')
319         end
320
321         if TO_current < 0
322             feedback = 0;
323             TO_current_index;
324             instep_counter = 0;
325             set(SjWin.editWarnings,'BackgroundColor','
                    yellow','String','Toe off event')
326
327             % search for MSW event - max in gyroY
328             if sample_count - TO_current_index < 0.5*
                    stride_t*100
329                 MSW_current = MaxGyroFct(diffvel_y,
                        diffvel_y_pre, gyro_y_buffer(1,2));
330             end
331
332             if MSW_current > 0 & MSW_switch == 1
```

```
333                         set(SjWin.editWarnings,'
                                BackgroundColor','yellow','String','
                                Mid-swing')
334                         % store value and index of max here
335                         MSWindex = sample_count+1;
336                         % output of data
337                         dataOut = [IC_current_index,
                                IC_current, TO_current_index,
                                TO_current, MSWindex, MSW_current,
                                feedback];
338
339                         %reset values to look for gait events
                                in
340                         %next stride
341                         IC_switch = 0;
342                         IC_current = 0;
343                         TO_current = 0;
344                         MSW_switch = 0;
345                         time_to_IC = sample_count;
346
347                     end
348                 end
349             end
350         else
351             % if subject has not started walking, output is
                    zero
352             dataOut = zeros(1,numOutputs);
353         end
354
355         % Graphics GyroY plot
356         buffy1(1,plotIndex) = dataGyroY;
357         set(SjWin.plot1,'Xdata',1:1:200,'Ydata', buffy1(1,:),'
                color','blue')
358
359         buffy2(1,plotIndex) = dataAccX;
360         set(SjWin.plot2,'Xdata',1:1:200,'Ydata', buffy2(1,:),'
                color','blue')
361
362         % Slide data points to the left to simulate plot
                generation in real time
363         if(plotIndex == length(buffy1) && plotIndex == length(
                buffy2))
364             plotIndex = 50;
365             buffy1(1,1:100) = buffy1(1,101:200);
366             buffy1(2,1:100) = buffy1(2,101:200);
367             buffy2(1,1:100) = buffy2(1,101:200);
368             buffy2(2,1:100) = buffy2(2,101:200);
```

```
369
370                     buffy1(1,101:200) = 0;
371                     buffy1(2,101:200) = 0;
372                     buffy2(1,101:200) = 0;
373                     buffy2(2,101:200) = 0;
374                 end
375             % If Start button is off.
376             else
377
378                 % Change colour of the button back to initial, to
                       inform on
379                 % the current state. Reinitialise graphics.
380
381                 if (controlStart == 1)
382                     controlStart = 0;
383                     set(SjWin.buttonStart,'BackgroundColor','black','
                           String','Start','ForegroundColor','white')
384                     buffy1(:) = 0;
385                     buffy2(:) = 0;
386                     set(SjWin.plot1,'Xdata',1:1:200,'Ydata', buffy1
                           (1,:),'color','black')
387 %                     set(SjWin.plot3,'Xdata',1:1:100,'Ydata', buffy1
    (2,:),'color','black')
388                     set(SjWin.plot2,'Xdata',1:1:200,'Ydata', buffy2
                           (1,:),'color','black')
389 %                     set(SjWin.plot4,'Xdata',1:1:100,'Ydata', buffy2
    (2,:),'color','black')
390                 end
391                 dataOut = zeros(1,numOutputs);
392         end
393
394         % index for two plots
395         plotIndex = plotIndex+1;
396
397
398 %% Button functioning
399
400 % Stop button
401 if(pushStop)
402     close(SjWin.figHandle);
403 end
404 % Save button
405 if(pushSave)
406     pushSave = 0;
407     set(SjWin.buttonSave,'BackgroundColor','white','String','Saving...'
          ,'ForegroundColor','black')
408
```

```
409     dataThrAccMax = str2num(get(SjWin.editUmbralTalonPareticoL1,'String
            '));
410     dataThrAccMin = str2num(get(SjWin.editUmbralPuntaPareticaL1,'String
            '));
411     dataThrGyroMax = str2num(get(SjWin.editUmbralTalonPareticoL2,'
            String'));
412     dataThrGyroMin = str2num(get(SjWin.editUmbralPuntaPareticaL2,'
            String'));
413
414     conf = [dataThrAccMax,dataThrAccMin,dataThrGyroMax,dataThrGyroMin,
            dataStride];
415     save ('gaitDetection.mat','conf');
416     pause(3)
417     set(SjWin.buttonSave,'BackgroundColor',[.94 .94 .94],'String','Save
            ','ForegroundColor','black')
418 end
419
420 % Start button
421 if(pushStart)
422      pushStart = 0;
423      if(start == 0)
424          start = 1;
425      else
426          start = 0;
427      end
428 end
429
430 %% Output data
431
432     dataout(numOutputs)= toc;
433     sys = [dataOut];
434
435     %------------%
436     % Terminate %
437     %------------%
438     case 9
439      sys=mdlTerminate(t,x,u);
440
441     case {1, 2, 4}
442      sys=[];
443
444     %%%%%%%%%%%%%%%%%%%
445     % Unexpected flags %
446     %%%%%%%%%%%%%%%%%%%
447     otherwise
448      DAStudio.error('Simulink:blocks:unhandledFlag', num2str(flag));
449
```

```
450 end
451
452 % end sfuntmpl
453
454 %
455 %
    ================================================================================

456 % mdlInitializeSizes
457 % Return the sizes, initial conditions, and sample times for the S-
    function.
458 %
    ================================================================================

459 % [ not shown here]
```

10.2 Offline event detection

```
1  filename = 'C:\Users\ioana\Desktop\Uni\MEng Project\FSR\FSR+IMU data\
       sub020.capp';
2  %filename = "C:\Javi\Universidad\Practicas Instituto Cajal\MATLAB\
       Codigos Subject 001\Subject 001 - copia\001_IMU\CAPTURE_MCS9_1
       -1-2000.capp";
3  [acc,angles,magn,temp] = import_physical_data_Ina(filename, 1, 11813);
       %,[25,35]);
4
5  IMU = 1;
6  shankaccx = acc{1,1}; %take data from cell into list
7  shankaccy = acc{3,1};
8  shankaccz = acc{2,1};
9  shankanglex = angles{1,1};
10 shankangley = angles{3,1};
11 shankanglez = angles{2,1};
12
13 Fs = 100; %sampling frequency
14 time = length(shankaccx)/Fs; %
15 cadence = n*60/time;
16 steptime = int16(length(shankaccx)/n);  % in number of samples
17
18 maxvel = []; % mid-swing
19 maxvelind = [];
20 minvel = [];
21 minvelind = [];
```

```
HS = [];
HSindex = [];
TO = [];
TOindex = [];
diffvel = diff(cleanangle);
maxthresh = 1; % value from paper
minthresh = -1.5;
stride_len = [];
HSprev_index = 0;
TOprev_index = 0;
MSWprev_index = 0;
for i = 2 : (length(shankangley)-1)
    if diffvel(i) < 0 &&  diffvel(i-1) > 0 && cleanangle(i) >
       maxthresh && i - MSWprev_index > 100
        maxvel = [maxvel, cleanangle(i)];
        maxvelind = [maxvelind, i];
        MSWprev_index = i;
    else
        if diffvel(i) > 0 &&  diffvel(i-1) < 0 && shankangley(i) <
           minthresh
            minvel = [minvel, cleanangle(i)];
            minvelind = [minvelind, i];
        end
    end
end
n = length(maxvel);
counter = 1;
% use data about gait phase duration from literature: IC at 0% step
   cycle,
% MSW between 75-87% gait cycle
first = maxvelind(1);
[M,I]=min(cleanangle(1:first));
TO = [M];
TOindex = [I];
for k = 1:(n-1)
    a = maxvelind(k);
    b = maxvelind(k+1);
%     c = HS(k);
    stride_len = [stride_len, b-a];
    for j = a:b
        if diffvel(j)<0 && diffvel(j-1)>0 && j>a+13*(b-a)/100 && j<a
           +25*(b-a)/100 && cleanangle(j)>0
        %if diffvel(j)<0 && diffvel(j-1)>0 && shankangley(j)== max(
           shankangley(a+5:b-10)) % can't be max
            p2 = [p2, cleanangle(j)];
            p2index = [p2index, j];
        end
```

```
64          if diffvel(j)>0 && diffvel(j-1)<0 && j<a+30*(b-a)/100 &&
               cleanangle(j)<0 & j - HSprev_index > 0.7*(b-a) % need a
               motivation for interval here
65              HS = [HS, cleanangle(j)];
66              HSindex = [HSindex, j];  % bogus detections
67              HSprev_index = j;
68          end
69          if diffvel(j)>0 && diffvel(j-1)<0 && cleanangle(j)<minthresh &
                j - TOprev_index > 0.7*(b-a) & j > a+50*(b-a)/100
70              %&& j - TOindex(counter)>mean(stride_len) - 2*std(
                   stride_len)
71              TO = [TO, cleanangle(j)];
72              TOindex = [TOindex, j];
73              counter = counter + 1;
74              TOprev_index = j;
75 %               if TOindex(counter) - TOindex(counter-1) < mean(
       stride_len) - 2*std(stride_len)
76 %                   TOindex(counter-1) = [];
77 %                   TO(counter-1) = [];
78 %                   counter = counter - 1;
79 %               end
80          end
81      end
82  end
```

10.3 Eurobench literature review Scopus search string

Search 1 keywords and wildcards

TITLE-ABS-KEY (locomot* OR gait* OR walk* OR "body transport*") AND TITLE (test* OR assess* OR measure* OR benchmark* OR evaluat*) AND TITLE (prosth* OR "artificial limb") AND TITLE-ABS-KEY (knee OR transfemoral OR foot OR ankle OR transtibial OR leg OR "lower-limb" OR "lower-extremity" OR "lower-leg") AND TITLE-ABS-KEY (active OR adaptive OR artificial OR biomechatronic OR biomimetic OR bionic OR intelligent OR powered) AND NOT TITLE-ABS-KEY (replacement OR arthroplast*)).

Search 2 keywords and wildcards

(TITLE-ABS-KEY (locomot* OR gait* OR walk* OR "body transport*") AND TITLE-ABS-KEY (test* OR assess* OR measure* OR benchmark* OR evaluat*) AND TITLE (prosth* OR "artificial limb") AND TITLE-ABS-KEY (knee OR transfemoral OR foot OR ankle OR transtibial OR leg OR "lower-limb" OR "lower-extremity" OR "lower-leg") AND TITLE-ABS-KEY (active OR powered)).